

Integració de la gestió de projectes d'R+D+I dins el portal RuralCat

Grau Enginyeria Informàtica

Enginyeria del Software

29/01/2019

Directora

Maria Dolores Villanueva

Ponent

Dolors Costal

Autor

Albert Sabaté

Resum

En un món globalitzat, cada cop més, els projectes superen les fronteres i aglutinen actors de diferents països. La Generalitat de Catalunya i el seu sector agroalimentari participen en aquest fenomen amb la voluntat de ser part activa i destacada del futur d'Europa i de la Mediterrània.

RuralCat vol construir un espai web per donar cabuda a les iniciatives europees i internacionals que es recolzen o es duen a terme des del Departament d'Agricultura, Ramaderia, Pesca i Alimentació de la Generalitat de Catalunya. El propòsit és facilitar la informació dels projectes als ciutadans per tal donar a conèixer els diferents projectes que impulsa la Unió Europea.

D'altra banda, també es vol facilitar la gestió de la informació. Unificant-la tota sota un mateix espai web i seguint un mateix format. De tal manera que sigui més senzill mostrar nous projectes com actualitzar la informació dels ja existents.

Abstact

In a globalized world, more and more, projects exceed borders and bring together actors from different countries. The *Generalitat* of Catalonia and its agro-food sector participate in this phenomenon with the intention of being an active and prominent part of the future of Europe and the Mediterranean.

RuralCat wants to build a web space to accommodate the European and international initiatives that are supported or carried out by the Department of Agriculture, Livestock, Fishing and Food of the *Generalitat* de Catalunya. The purpose is to facilitate the information of the projects to the citizens in order to publicize the different projects promoted by the European Union.

On the other hand, we also want to facilitate the management of the information. Unifying it all under the same web space and following the same format. In such a way that it is easier to show new projects such as updating the existing information.

Glossari

CMS: software de gestió de continguts, que permet elaborar-los, publicar-los i actualitzar-los. S'utilitza especialment per a gestionar pàgines web

CSV: software que manté el registre de tot el treball i de tots els canvis en els fitxers del projecte. Permet que diferents desenvolupadors col·laborin de manera eficient.

Framework: marc de treball que facilita la concepció de les aplicacions mitjançant la utilització de biblioteques de classes o generadors de programes.

HTTP: protocol de comunicació que permet la transferència d'informació a Internet

Llenguatge interpretat: són aquells que necessiten un programa intèrpret per a ser executats. Les instruccions que descriuen els programes o utilitats desenvolupats amb aquests no són instruccions que entén directament la plataforma que el conté i per tant necessiten un programa que l'interpreti.

Plugins: aplicació que es relaciona amb un sistema per afegir una nova funcionalitat, generalment molt específica.

Scrum: és un marc de treball per a la gestió de projectes. L'objectiu és desenvolupar i crear un producte en un període determinat on un equip format per diferents persones treballen conjuntament per arribar a un objectiu comú.

Stakeholder: és l'equivalent en anglès a les parts interessades en un projecte, les quals busquen obtenir un benefici amb la realització del projecte.

Target: persona o grup de persones a qui va dirigit un producte.

Tuples: estructura que representa un conjunt únic de dades.

Waterfall Model: és un procés de disseny seqüencial en què el progrés flueix constantment cap avall (com una cascada) a través de les diferents fases.

Índex de continguts

1.	Introducció	11
1.1.	Context del projecte	11
1.2.	Actors implicats	12
1.2.1.	Adasa Sistemas.....	12
1.2.2.	Comsa.....	12
1.2.3.	Generalitat de Catalunya.....	12
1.2.4.	Públic objectiu.....	12
1.3.	Estat de l'art.....	13
1.4.	Descripció del problema.....	15
1.4.1.	Objectius principals.....	16
1.4.2.	Objectius secundaris	17
1.5.	Abast.....	17
1.5.1.	Riscos.....	17
1.6.	Metodologia	18
1.6.1.	Eines de seguiment	19
2.	Planificació temporal.....	20
2.1.	Calendari.....	20
2.2.	Recursos.....	20
2.2.1.	Humans	20
2.2.2.	Materials	21
2.3.	Descripció de les tasques	22
2.3.1.	Inepció	23

2.3.2.	Elaboració.....	23
2.3.3.	Construcció.....	23
2.3.4.	Transició	24
2.3.5.	Fase final del projecte	24
2.3.6.	Desglossament de les tasques	25
2.4.	Pla d'acció i valoració d'alternatives	25
2.5.	Diagrama de Gantt	27
3.	Gestió econòmica.....	28
3.1.	Identificació i estimació de costos.....	28
3.1.1.	Estimació de costos.....	28
3.1.2.	Costos directes	28
3.1.3.	Costos indirectes	31
3.1.4.	Cost total	32
4.	Especificació de requisits	34
4.1.	Requisits funcionals.....	34
4.1.1.	Diagrames de casos d'ús	34
4.1.2.	Especificació casos d'ús.....	36
4.1.3.	Model conceptual de les dades	46
4.1.4.	Descripció de les classes	47
4.2.	Requisits no funcionals.....	50
5.	Arquitectura	54
5.1.	Visió global	54
5.2.	Diagrama de classes de disseny	55

5.3.	Patrons de disseny	56
5.4.	Exemples de diagrames de seqüència.....	58
5.5.	Disseny de la interfície.....	61
5.6.	Bases de dades	65
5.6.1.	Disseny lògic.....	65
5.6.2.	Disseny físic	66
6.	Implementació	69
6.1.	Tecnologies i llenguatges emprats.....	69
6.2.	Eines de desenvolupament	72
7.	Proves.....	73
8.	Aspectes Legals	74
8.1.	Lleis aplicables al projecte	74
8.2.	Llicències	75
9.	Seguiment del projecte	77
9.1.	Canvis de planificació i pressupost	77
10.	Sostenibilitat	79
10.1.	Dimensió ambiental.....	80
10.2.	Dimensió econòmica	83
10.3.	Dimensió social	84
10.4.	Matriu de sostenibilitat	85
11.	Conclusions i treball futur	86
11.1.	Competències tècniques treballades i relació amb l'enginyeria del software	86
11.2.	Conclusions personals.....	88

11.3. Treball futur	89
12. Referències.....	90

Índex de figures

Figura 1. Organització dels usuaris.....	16
Figura 2. Waterfall Model.....	19
Figura 3. Unified Process.....	19
Figura 4. Diagrama de Gantt	27
Figura 5. Model conceptual de les dades.....	46
Figura 6. Diagrama de classes de disseny	55
Figura 7. Patró MVC	57
Figura 8. Patró Singleton	57
Figura 9. Diagrama de seqüència: alta projecte.....	59
Figura 10. Diagrama de seqüència: cercar Projecte.....	60
Figura 11. Interfície cercador	61
Figura 12. Interfície visualitzar projecte.....	62
Figura 13. Interfície crear/editar projecte.....	63
Figura 14. Interfície dashboard	64
Figura 15. Disseny lògic	66
Figura 16. Disseny físic	67

Índex de taules

Taula 1. Estat de l'art	14
Taula 2. Recursos materials	22
Taula 3. Desglossament de les tasques	25
Taula 4. Costos directes (Cas real)	29
Taula 5. Costos directes (Cas hipotètic)	31
Taula 6. Costos indirectes	32
Taula 7. Costos totals	33
Taula 8. Llicències	76
Taula 9. Desviacions en la planificació (hores i pressupost)	78
Taula 10. Desviacions en la planificació (dates)	78
Taula 11. Model matriu de sostenibilitat	79
Taula 12. Consum energètic	81
Taula 13. Estalvi energètic	82
Taula 14. Matriu de sostenibilitat	85

1. Introducció

1.1. Context del projecte

Aquest projecte és un Treball Final de Grau en l'especialitat d'Enginyeria del Software de la Facultat d'Informàtica de Barcelona (Universitat Politècnica de Catalunya). És un projecte cursat en la modalitat B en l'empresa Adasa Sistemas [1].

Adasa Sistemas, actualment membre de Comsa [2], és una empresa especialitzada a donar solucions tecnològiques per a problemes relacionats amb l'aigua, el medi ambient i la meteorologia. És una empresa internacional amb oficines a Espanya, Marroc i Austràlia. A més a més, mitjançant un contracte públic amb la Generalitat de Catalunya, Adasa proporciona i dona suport a un portal web, RuralCat [3].

RuralCat és un portal creat amb el gestor de continguts (CMS) Liferay [4] i està dirigit al món rural, actuant, en molts casos, d'intermediari entre la població i el departament d'Agricultura, Ramaderia, Pesca i Alimentació. Aquest portal té un volum d'usuaris de gairebé 40 milers de persones, el qual pot semblar poc, però és molt alt si el comparem amb la població que es dedica al sector de l'agricultura (el públic al qual va dirigit el portal), 60 milers de persones aproximadament a Catalunya en el 2018 [5]. S'encarrega de fer difusió dels diferents avisos i notícies que envia la Generalitat i gestiona la inscripció dels cursos oficials relacionats amb el món rural. Per tant, per la gent dedicada al camp, RuralCat és un portal de referència.

Dintre el portal conviuen diferents espais web independents de la resta del contingut de RuralCat. El disseny i el contingut d'aquests espais web és dictat estrictament per agents externs (el client), però gestionat sota petició des d'Adasa. La Xarxa d'innovació agroalimentària i rural de Catalunya és un d'aquests espais i el client d'aquest projecte. És el principal canal de difusió dels resultats de les iniciatives i projectes en l'àmbit agroalimentari català. Actua com a element dinamitzador de la innovació dels sectors agroalimentari i forestal, com a facilitador de coneixements entre la ciència i la pràctica i com a element que afavoreix la transferència de coneixements i oportunitats entre explotacions, indústries, investigadors i tècnics del sector agroalimentari.

1.2. Actors implicats

En aquest apartat es definiran els actors principals (*stakeholders*) d'aquest projecte. Aquests són les persones i les entitats que estan relacionades amb el projecte, sigui perquè participen en el desenvolupament o perquè són el públic a qui va dirigit el del producte.

1.2.1. Adasa Sistemas

Adasa és l'empresa responsable del portal RuralCat, en el qual s'ha d'integrar el nou software. Per aquest motiu participarà activament en el desenvolupament del projecte i serà l'encarregada de la comunicació amb el client.

1.2.2. Comsa

Comsa és el grup empresarial al qual pertany Adasa. En un principi no té cap relació amb el projecte però està interessada en què es desenvolupi correctament, ja que el contracte públic amb la Generalitat, en el qual s'adjudica a Adasa el desenvolupament i manteniment de RuralCat, és una part important d'una de les empreses que formen Comsa.

1.2.3. Generalitat de Catalunya

La Generalitat de Catalunya, concretament el departament d'Agricultura, Ramaderia, Pesca i Alimentació (Xarxa d'innovació agroalimentària i rural de Catalunya), és el client del projecte. Per tant, s'encarrega de definir el problema a resoldre i té un gran interès en poder donar-li una solució satisfactòria.

1.2.4. Públic objectiu

El públic objectiu són aquelles empreses i organitzacions, tant públiques com privades, a qui va destinat el projecte. Per tant, aquest col·lectiu serà el qui doni ús al sistema, ja sigui per gestionar els seus projectes com per consultar els dels altres. També és possible que algun particular estigui interessat a consultar la informació d'algun projecte, però no és principal públic del projecte.

1.3. Estat de l'art

És molt important en aquest projecte decidir quin CMS s'utilitzarà per desenvolupar el sistema, ja que és la tecnologia principal que s'utilitzarà durant la implementació.

Primerament s'ha d'entendre perquè només es planteja l'opció d'utilitzar un CMS. El motiu és que les persones que en molts casos actualitzen o canvien la informació del portal no tenen alts coneixements informàtics, i els CMS tenen l'avantatge de ser molt senzill d'utilitzar un cop el sistema ha estat implementat. Per aquest motiu cal donar un entorn fàcil d'utilitzar per no generar problemes en el futur.

Un cop decidit que utilitzarem un CMS, hem de decidir quin utilitzarem. El principal candidat, i el que finalment s'ha escollit, és Liferay, ja que el portal on s'integrarà el projecte ja utilitza aquest gestor de continguts. Tot i això, atès que és una tecnologia crítica per al sistema, cal analitzar quines alternatives tenim: WordPress, Joomla, Drupal i Liferay.

Aquests gestors comparteixen molts trets comuns: flexibilitat per crear qualsevol tipus de web, facilitat en l'actualització i la modificació de la informació, compatibilitat amb els navegadors més utilitzats, bona gestió del posicionament en els buscadors (com per exemple Google), alta personalització de l'aparença de la web, alta escalabilitat de la web i actualitzacions molt freqüents. La principal diferència entre aquestes tecnologies és quin balanç proporcionen respecte a la senzillesa d'ús i la creació de sistemes complexos. Per aquest motiu l'elecció ideal del CMS que utilitzarem hauria de ser el més senzill dels que poden abraçar la complexitat del software que es vol desenvolupar.

WordPress [6] és el gestor de continguts més utilitzat al mercat, principalment per la seva facilitat en l'ús. Això el fa especialment indicat per a webs molt senzilles i per usuaris principiants. A causa d'aquesta gran quantitat d'usuaris, la comunitat de WordPress és molt extensa, i pot ser molt útil per resoldre qualsevol problema. El principal problema de WordPress, però, és que té dificultat per gestionar webs complexes.

Joomla [7], a l'igual que WordPress, també és molt senzill d'utilitzar, tot i que té una corba d'aprenentatge major. L'aspecte clau que diferencia a Joomla de la resta és la versatilitat dels seus

components, els quals poden canviar completament la seva aparença i funcionalitat. Per culpa d'això Joomla requereix major qualificació en els desenvolupadors, però també pot assolir webs més complexes.

Drupal [8] és un gestor més complex que WordPress i Joomla Tot i això, aquesta complexitat està compensada amb una major capacitat per construir webs grans sense necessitat de *plugins* externs. A més a més té integrades una sèrie d'eines, que els anteriors CSMs no incorporen, que fan de Drupal un software molt complet.

Liferay és el gestor més complex i complet dels que hem vist fins ara i per tant el que pot gestionar les webs més complexes. Per això es requereix desenvolupadors amb bons coneixements informàtics si es vol desenvolupar funcionalitats específiques. Està basat en el llenguatge de programació Java, això fa que sigui molt més fàcil gestionar la seguretat davant d'errors que en la resta de CMS que hem vist, els quals estan basats en PHP.

CMS	Corba d'aprenentatge	Complexitat	Seguretat	Llenguatge
Wordpress	3	2	1	PHP
Jomla!	3	2	1	PHP
Drupal	2	3	1	PHP
Liferay	1	3	3	Java

Taula 1. Estat de l'art

Realment és probable que el sistema sigui possible d'implementar en qualsevol de les quatre tecnologies vistes. Tot i això, davant el dubte, el millor és assegurar i triar entre Drupal i Liferay. Ambdues tecnologies tenen els seus avantatges i els seus inconvenients, Drupal és més senzill d'aprendre mentre que Liferay gestiona millor la seguretat. Però el fet que el portal on s'ha

d'integrar el sistema, Ruralcat, ja utilitzi Liferay, decanta la balança a favor d'aquest CMS, ja que la integració serà molt més senzilla.

1.4. Descripció del problema

RuralCat, com a portal del departament d'Agricultura, Ramaderia, Pesca i Alimentació, ha de construir un espai web per donar cabuda a les iniciatives europees i internacionals finançades amb fons públics amb la voluntat d'afavorir la difusió dels resultats obtinguts, potenciar el sistema d'innovació agroalimentària i accelerar la implantació en el teixit productiu dels coneixements generats. Per aquest motiu RuralCat rep informació de diferents projectes relacionats amb el món rural i finançats amb fons públics europeus amb la finalitat de fer-ne difusió.

El principal problema que afronta RuralCat és que fins ara han rebut tota la informació dels projectes directament des d'una base de dades externa a l'empresa i, per tant, han hagut d'adaptar-se al format de les dades que rebien (el qual consisteix en una sola taula SQL, indiferentment de si aquestes dades fan referència a diferents conceptes). Per poder construir el nou espai web necessiten poder reestructurar el format de les dades per tal de fer-les més entenedores i poder treballar de manera més eficaç. També necessiten poder introduir la informació dels projectes directament des del portal, per així reduir les comunicacions supèrflues entre Adasa i la Generalitat de Catalunya.

En segon lloc, és necessari mostrar tota la informació dels projectes guardats. Actualment existeix una llista amb el nom de tots els projectes des d'on es pot accedir a part de la informació de cada un dels projectes. El problema és que aquesta manera de visualitzar els projectes és insuficient. Actualment, com que hi ha un número reduït d'informació, es pot arribar a trobar un projecte concret, però en el futur és molt possible que el volum d'informació escali a nivells superiors. Per tant és necessari trobar una millor manera de poder accedir a la informació de cada un dels projectes.

Finalment la necessitat d'un espai web on es pugui introduir informació important planteja un últim problema, no tots els usuaris poden estar autoritzats a utilitzar aquest espai, sinó només aquells usuaris que indiqui expressament el departament d'Agricultura, Ramaderia, Pesca i

Alimentació. Cada un d'aquests usuaris amb "privilegis" només podrà modificar la informació dels seus projectes o dels projectes del seu grup d'usuaris (un grup d'usuaris representaria una entitat, com per exemple, una universitat o una empresa). Donades aquestes circumstàncies es fa patent la necessitat de crear un sistema de gestió de la informació adequat a les necessitats descrites.

1.4.1. Objectius principals

Els objectius principals d'aquest projecte són:

- Reestructurar la base de dades existent a una més entenedora i senzilla per poder treballar.
- Crear un espai web on es pugui introduir al portal tota la informació d'un projecte o modificar la informació d'un projecte ja existent.
- Crear un espai web on es pugui cercar els projectes en funció de la seva informació.
- Crear un espai web on es pugui visualitzar tota la informació d'un projecte.
- Crear un sistema de rols i grups d'usuari (vegeu Figura 1) per tal que:
 - Només els usuaris amb un rol específic puguin introduir i modificar informació dels projectes.
 - Els usuaris no puguin modificar la informació de projectes que no pertanyen al seu grup d'usuaris.

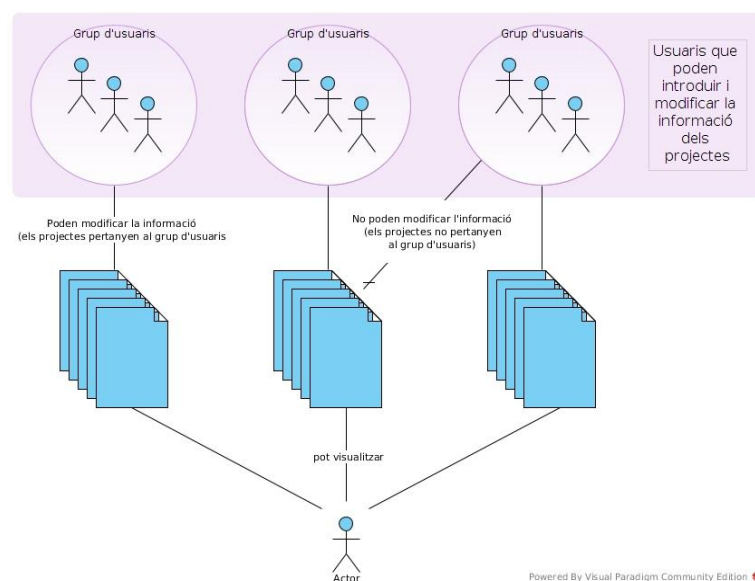


Figura 1. Organització dels usuaris

1.4.2. Objectius secundaris

Els objectius principals donen una solució efectiva al problema plantejat, tot i això, hi ha una sèrie d'objectius menors, que si ve no són crítics, és important assolir-los per tal d'aconseguir un software que compleixi amb totes les expectatives:

- Dissenyar la interfície del nou subsistema en concordança amb la resta del portal.
- Dissenyar una interfície intuïtiva i fàcil d'utilitzar.
- En la mesura del possible, la informació dels projectes es mostrarà de forma compacta, per tal que es pugui imprimir en el mínim de fulls possibles.
- Analitzar els recursos existents al portal i reutilitzar-los sempre que es pugui per minimitzar el cost.

1.5. Abast

L'abast d'aquest projecte ve donat per l'assoliment dels objectius que es volen assolir i pels possibles inconvenients o riscos que puguin haver-hi durant el desenvolupament del sistema. Per tal d'assolir els objectius plantejats podem dividir el projecte en tres parts diferenciades.

En primer lloc cal analitzar i estudiar els recursos utilitzats en el portal RuralCat. En la mesura del possible s'intentarà utilitzar els mateixos recursos o algun de similar per tal d'afavorir una integració coherent entre el nou sistema i el ja existent.

En segon lloc és necessari triar les eines que s'utilitzaran per desenvolupar el projecte utilitzant d'informació recollida anteriorment. Si és possible, s'intentarà reutilitzar algun dels recursos ja existents.

Finalment es desenvoluparà el sistema en la seva totalitat, des de l'anàlisi dels requisits fins a implementar el sistema.

1.5.1. Riscos

Durant el transcurs del projecte hi ha certs riscos que poden alterar l'abast del projecte. Per tant és important preveure'ls per tal de poder gestionar-los correctament.

- El client podria plantejar nous problemes. Això significaria augmentar el nombre de requisits del sistema i per tant el volum de feina ha realitzar.
- És molt probable que algunes de les tecnologies siguin desconegudes a l'iniciar el projecte. Per tant el desenvolupament del sistema resultant serà simultani a l'aprenentatge d'aquestes eines. Això pot alentir el desenvolupament en funció de la dificultat per realitzar aquesta tasca
- El temps del projecte és limitat perquè és un treball de fi de grau. Tot i que en un principi s'ha reservat temps de sobra cal tenir en compte aquest factor.

1.6. Metodologia

Per a desenvolupar un projecte de software existeixen diferents metodologies de treball, les quals podem diferenciar entre clàssiques (com per exemple el famós *Waterfall Model*) i àgils (de les que podem destacar *Scrum*). El millor llavors és escollir la metodologia en funció de quina s'adapta millor a les necessitats del projecte.

Per poder escollir correctament hem d'analitzar en quines circumstàncies es desenvoluparà el sistema. Tot el procés fins a arribar a posar en producció el sistema estarà desenvolupat únicament per una persona, per tant no és necessària cap mena de comunicació entre programadors (com les reunions diàries de *Scrum*). També cal tenir en compte que els requeriments del sistema ja han estat completament definits i el client no té cap previsió d'ampliar-los.

Tenint en compte aquesta informació s'ha decidit descartar les metodologies àgils, que si bé poden ser molt útils en molts casos, no s'adapten correctament al projecte. Són metodologies més recomanables per equips petits de programadors, en sistemes amb requeriments en constant canvi i on és important proporcionar software útil en cada iteració.

Un cop escollit les metodologies clàssiques cal decidir quina d'elles utilitzar: les més conegudes són el *Waterfall Model* i l'*Unified Process*. La principal diferencia és que en el *Waterfall Model* [9] cada fase (anàlisis, disseny, implementació, proves, etc.) es realitza de forma seqüencial (vegeu figura 2), és a dir, no es comença una fase fins que no s'ha finalitzat l'anterior (per exemple, no es pot començar a implementar fins que els requisits hagi estat completament identificats i el disseny

estigui aprovat). D'altra banda en l'*Unified Process* [10] les fases es realitzen de manera iterativa i concurrentment (vegeu figura 3). Cada fase és multidisciplinària, per exemple, a la fase d'Incepció, que es centra en l'anàlisi dels requeriments, es pot començar a dissenyar el sistema.

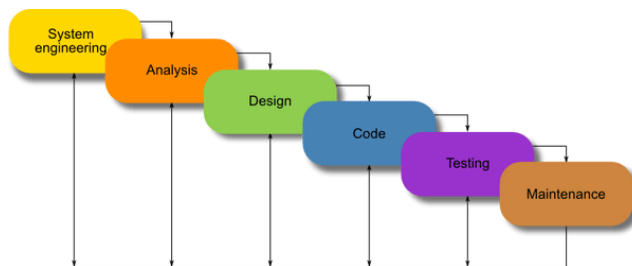


Figura 2. Waterfall Model

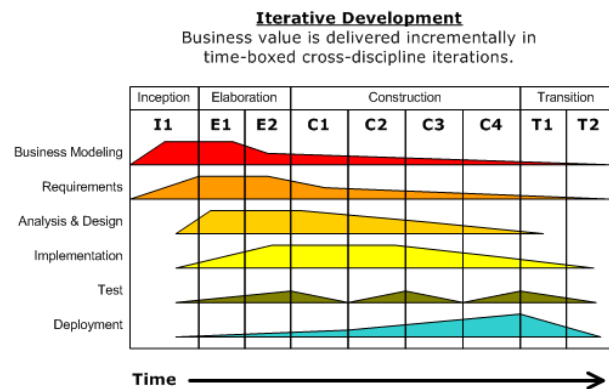


Figura 3. Unified Process

La metodologia a utilitzar en el projecte serà l'*Unified Process* (UP), ja que té una major flexibilitat que el *Waterfall Model* i permet gestionar els riscos amb més facilitat (gràcies a les diverses iteracions i al *feedback* resultant d'aquestes).

1.6.1. Eines de seguiment

A causa del fet que el projecte estarà desenvolupat únicament per una persona no és necessari tenir cap CSV ni tampoc cap sistema d'integració continua. Tot i això, per poder tenir un registre de tot el treball fet, sí que he decidit utilitzar *Git*. Per gestionar les tasques pendents s'utilitzarà *Trello*, és una eina poc complexa però que es pot adaptar molt bé en situacions senzilles. Finalment per comunicar-me amb la directora del projecte i poder tenir un seguiment s'utilitzarà en gran mesura *Skype* i en menor mesura el correu electrònic.

2. Planificació temporal

2.1. Calendari

El projecte té una durada, en la seva totalitat, de set mesos, començant el 2 de juliol (el dia que m'incorpo a l'empresa) i acabant a finals de gener (data aproximada de la defensa oral). Tot i això, el procés de desenvolupament del projecte tindrà una durada aproximada de quatre mesos, començant a principis de juliol i acabant a finals d'octubre. Per tant hi ha un gran marge per possibles desviacions en la planificació.

Per poder preparar la defensa i acabar la memòria s'ha cregut convenient acabar el conveni entre finals de desembre i principis de gener, d'aquesta manera obtindrè una disponibilitat total per poder enllestir tota la feina restant de forma satisfactòria.

2.2. Recursos

Per dur a terme el projecte es necessitaran recursos de diferents àmbits. Podem classificar aquests en els següents grups: humans, materials i software. S'han diferenciat el que és purament material dels que són un subconjunt com són recursos de software, ja que en aquest cas com que és un projecte d'enginyeria informàtica tenen molt de pes per si mateixos.

2.2.1. Humans

L'únic recurs humà requerit és el desenvolupador que dissenyarà i desenvoluparà el sistema. Tindrà una dedicació de 30 hores setmanals, és a dir, de 6h diàries. Tot i això s'han de sumar les hores addicionals dedicades a la documentació del projecte i a la preparació de la defensa, que es realitzaran fora del lloc de treball.

2.2.2. Materials

Recurs	Tipus	Finalitat
Lloc de treball	Entorn de treball	Donar un espai adequat per al correcte desenvolupament del projecte
Ordinador sobretaula	Eina de desenvolupament	S'utilitzarà principalment per al desenvolupament del sistema.
Ordinador portàtil	Eina de desenvolupament	Ordinador on es realitzarà la memòria fora del lloc de treball
Servidor	Entorn de treball	Servidor on es desplegarà el sistema
Material d'oficina	Altres	Material necessari per poder prendre notes i realitzar esborranys
<i>Eclipse</i>	Eina de desenvolupament	Entorn de desenvolupament (IDE) on es desenvoluparà el sistema
<i>MySQL</i>	Eina de desenvolupament	Sistema de gestió SQL que s'utilitzarà per al desenvolupament del sistema
<i>Apache</i>	Eina de desenvolupament	Servidor HTTP per a desplegar el sistema
Liferay	Eina de desenvolupament	Gestor de continguts (CMS) que s'utilitzarà per desenvolupar el sistema

Software d'ofimàtica	Eina de desenvolupament	Software per realitzar la memòria del projecte i el material de suport per a la defensa oral
<i>Trello</i>	Eina de desenvolupament	Gestor de tasques per organitzar la feina durant el desenvolupament del projecte
<i>Outlook</i>	Eina de comunicació	Servei de correu electrònic utilitzat per a la comunicació amb la cap de projecte
<i>Skype</i>	Eina de comunicació	Software per a la comunicació amb la cap de projecte

Taula 2. Recursos materials

2.3. Descripció de les tasques

Tal com s'ha esmentat prèviament, el projecte comença el 2 de juliol i acabarà a finals de gener. En aquest apartat es descriuran i es planificaran les tasques a realitzar durant aquest interval de temps.

Abans de procedir a descriure les tasques que s'han de dur a terme durant el desenvolupament cal tenir en compte una sèrie de circumstàncies. Com ja s'ha esmentat prèviament, el projecte es realitzarà amb la metodologia *Unified Process*. Per tant el projecte es dividirà en quatre fases (Incepció, Elaboració, Construcció i Transició) i cada una d'aquestes fases pot tenir una o varies iteracions. A més a més, com que el projecte únicament estarà desenvolupat per una sola persona, i per tant les tasques de desenvolupament no es podran paral·lelitzar en cap cas, no s'ha cregut necessari realitzar un diagrama de PERT [11].

També és important recalcar que les 30 hores setmanals treballades a l'empresa van destinades exclusivament al desenvolupament del sistema. Mentre que totes les tasques de GEP, així com la realització de la memòria i la preparació de la defensa es realitzarà en fora de l'horari laboral.

2.3.1. Incepció

Iteració única (2 de Juliol – 13 de Juliol)

La fase d'Incepció coincideix en gran mesura amb la feina a realitzar a GEP. S'ha d'estudiar la viabilitat del projecte, realitzar un *planning* de la feina a fer i definir l'abast del projecte. També es començarà a analitzar els requisits que haurà de tenir el sistema. Finalment, com que aquesta fase coincideix amb la meua incorporació a l'empresa també s'haurà d'instal·lar tot el software necessari per poder desenvolupar el projecte.

2.3.2. Elaboració

Iteració única (16 de Juliol – 10 d'Agost)

Els objectius d'aquesta iteració són capturar la gran majoria dels requisits, definir l'arquitectura del sistema i crear els casos d'ús. Un últim objectiu també és donar solució als riscos crítics que en el cas d'aquest projecte consistirà a reestructurar la base de dades ja existent. Actualment la base de dades disposa únicament d'una taula SQL amb totes les dades, l'objectiu serà crear una nova estructura més entenedora i clara però sense destruir cap dada i mantenint la seva coherència.

2.3.3. Construcció

Aquesta és la fase que té una major durada en el desenvolupament del projecte. Estarà dedicada majoritàriament a desenvolupar el sistema segons els requeriments i els casos d'ús generats en iteracions anteriors. En cada iteració d'aquesta fase s'obtindrà un producte software llest per a ser utilitzat.

Primera iteració (13 d'Agost – 31 d'Agost)

En aquesta iteració es vol poder visualitzar de forma completa la informació dels projectes, poder modificar-la i poder introduir informació nova. Donat que són funcionalitats bastant similars, es té previst reutilitzar al màxim el codi, tant el de *frontend* com el de *backend* [3].

Segona iteració (3 de Setembre – 21 de Setembre)

En aquesta segona iteració es crearà el cercador de projectes, el qual podrà filtrar els projectes segons una sèrie de paràmetres (aquests paràmetres es definiran en el cas d'ús corresponent). Un cop obtinguts els projectes desitjats el sistema haurà de poder enllaçar amb les funcionalitats creades en la iteració anterior (visualitzar i modificar la informació d'un projecte).

Tercera iteració (24 de Setembre – 12 d'Octubre)

En aquesta iteració final es definiran els permisos per tal que només els usuaris indicats puguin crear projectes així com el sistema de grups d'usuari per tal que cada usuari només pugui modificar la informació dels projectes del seu grup. Per tal d'agrupar aquestes funcionalitats es crearà un *dashboard* on es podran visualitzar i editar els projectes propis. En finalitzar aquesta iteració el sistema ja hauria d'estar llest per poder passar a producció.

2.3.4. Transició

Iteració única (15 d'Octubre – 26 d'Octubre)

Durant aquesta iteració es desplega el sistema i s'obté el feedback final dels clients. De la mateixa manera es refina qualsevol requisit o funcionalitat que calgui. Si aquests refinaments són molt elevats en numero es podria plantejar l'opció de fer més d'una iteració en aquesta fase, però donada l'envergadura del projecte no s'ha cregut que sigui necessari.

2.3.5. Fase final del projecte

El desenvolupament del projecte està previst en les fases i en les iteracions corresponents, tot i això, un cop acabades, encara quedarà feina per finalitzar el projecte. Durant aquest temps restant es realitzarà la memòria utilitzant la documentació generada tant a GEP com en el transcurs de les diferents iteracions i també es prepararà la defensa.

2.3.6. Desglossament de les tasques

Tasca	Data inici	Data final	Dependent de	Hores
Incepció	02/07/18	13/07/18	---	60
Elaboració	16/07/18	10/08/18	Incepció	120
Construcció	13/08/18	12/10/18	Elaboració	270
Transició	15/10/18	26/10/18	Construcció	60
GEP	17/09/18	15/10/18	---	90
Memòria	23/07/18	23/11/18	Incepció	160
Defensa oral	26/11/18	25/01/19	Memòria	50
Total				810

Taula 3. Desglossament de les tasques

2.4. Pla d'acció i valoració d'alternatives

Durant el desenvolupament del projecte és possible que hi hagi algunes incidències que modifiquin la planificació temporal establerta. Tot i això, com que hi ha un gran marge per corregir aquestes desviacions (el projecte es preveu que estigui acabat a finals d'Octubre però es podria allargar el termini fins de Desembre) no hi hauria de perillar la viabilitat del projecte.

Aquestes possibles desviacions podrien venir donades per culpa d'una mala acotació del volum de feina a realitzar o pel fet que no es puguin realitzar les hores necessàries per realitzar el projecte.

En cas que el volum de feina no sigui l'esperat es poden donar dos casos: que el volum de feina sigui inferior. En aquest cas les iteracions seran menors del previst i s'acabarà el projecte abans del previst. En el cas contrari, que el volum sigui superior del previst, les iteracions es faran més llargues o, si el volum de feina realment es mol més gran, es realitzaran iteracions addicionals en la fase convenient.

2.5. Diagrama de Gantt

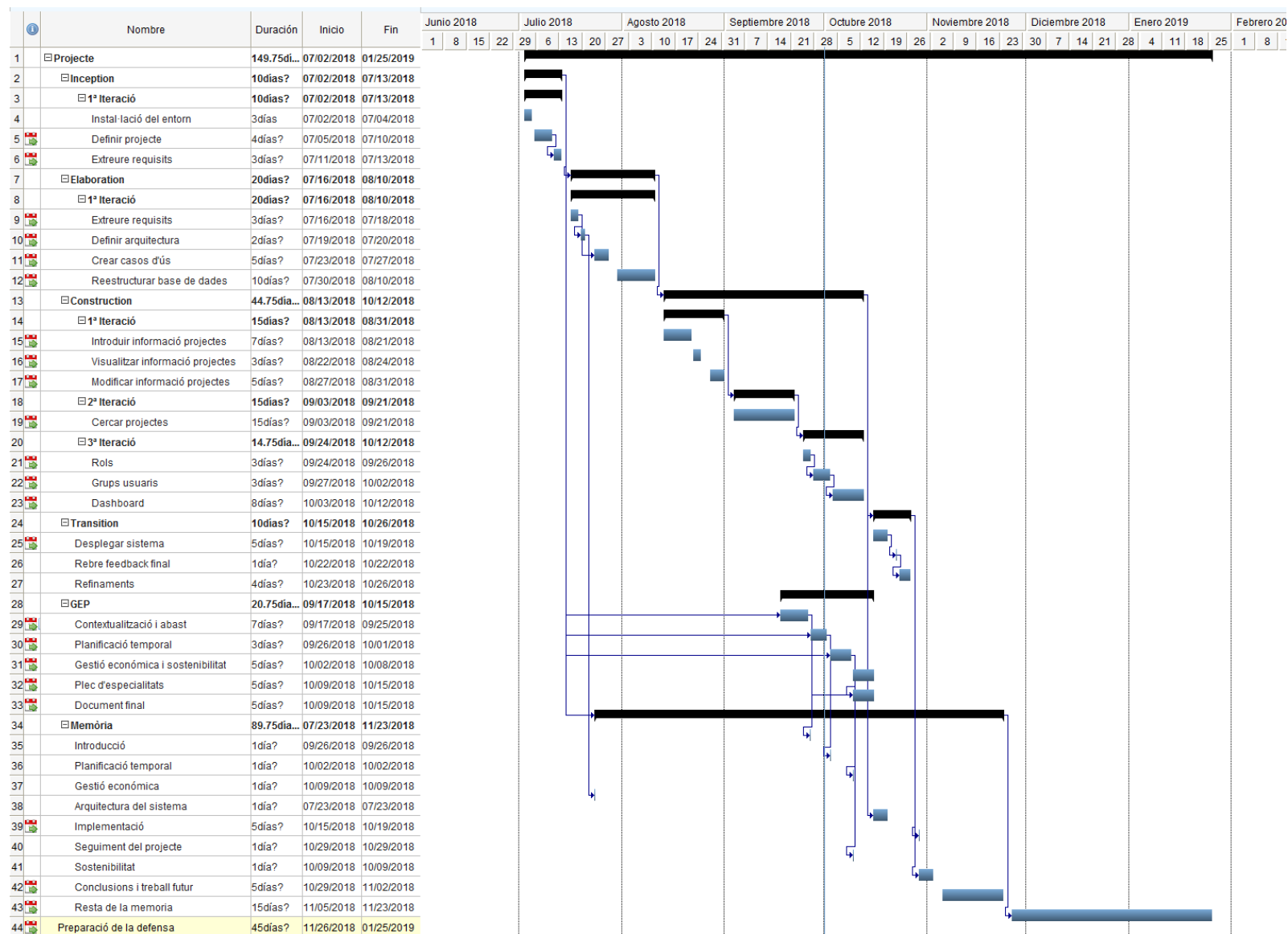


Figura 4. Diagrama de Gantt

3. Gestió econòmica

3.1. Identificació i estimació de costos

Una vegada planificat el projecte, és important realitzar un estudi dels costos que comportarà el desenvolupament del projecte. Aquestes dades ens serviran per fer un pressupost i visualitzar si el projecte és, o no, viable econòmicament.

3.1.1. Estimació de costos

Per a poder calcular els costos separarem els costos directes, els quals seran els derivats de la despesa dels sous dels treballadors i per tant en paguen en funció de les hores, dels costos indirectes, que únicament es paguen a l'adquirir-los (a part dels possibles costos de manteniment derivat del seu ús) o són costos que no es poden assignar a cap activitat.

3.1.2. Costos directes

Els costos directes són els costos imputables a l'execució de les activitats del diagrama de Gantt. Aquests costos variaran en funció de les hores de cada tasca i del preu a pagar per hora, depenent quin rol ocupi la persona que executa l'activitat.

Aquest projecte és un cas molt particular, ja que únicament estarà desenvolupat per una sola persona (un estudiant en condició de becari de pràctiques) que realitzarà totes les tasques i tots els rols. La remuneració serà de 8€/hora, aquesta es la tarifa mínima que estableix la UPC per a realitzar un conveni de pràctiques. L'empresa també ha de pagar un 15,7% del total a la UPC per a despeses de gestió, així com 40,13€ per a contingències comunes [12]. No es tindran en compte les hores treballades per a GEP, per la realització de la memòria ni per preparar la defensa del projecte, ja que, com que no estan remunerades, no tenen cap cost per a l'empresa. Per tan en el cas d'aquest projecte els costos directes serien, de forma molt simplificada, els següents:

Descripció	Hores	Cost (8 €/hora)	Total (€)
Hores realitzades al lloc de treball	510	8	4080
Despeses de gestió	---	---	640
Contingències comunes	---	---	40
Total			4760

Taula 4. Costos directes (Cas real)

Tot i això en un projecte real la feina estaria repartida entre diferents persones i rols. Per tant en un cas hipotètic més acord amb la realitat els costos directes serien molt diferents. En aquest projecte es podrien distingir quatre rols: Cap de projecte (CP) el qual s'encarrega de gestionar el projecte. En el mercat laboral a Espanya, aquest rol té una remuneració de 18 €/hora [13]. Analista (An), el qual defineix els requisits, en Espanya aquest rol pot tenir un salari de 20 €/hora [14]. Arquitecte (Ar), amb la tasca de dissenyar el sistema, amb un sou de 20 €/hora [15]. L'encarregat d'implementar el sistema segons l'especificat pel cap de projecte és el programador (P), el qual té un sou menor, 14 €/hora en el mercat laboral a Espanya [16]. En la següent taula podem veure un desglossament detallat dels costos directes d'aquest cas hipotètic

Activitat	Hores	Rol	Cost(€/hora)	Total(€)
Instal·lació de l'entorn	18	CP	18	324
Definir projecte	24	CP	18	432

Extreure requisits	36	An	20	720
Definir arquitectura	12	Ar	20	240
Crear casos d'ús	30	An	20	600
Reestructurar base de dades	60	P	14	840
Introduir informació projectes	42	P	14	588
Visualitzar informació projectes	18	P	14	252
Modificar informació projectes	30	P	14	420
Cercar projectes	90	P	14	1260
Rols	18	P	14	252
Grups usuari	18	P	14	252
Dashboard	48	P	14	672
Desplegar sistema	30	P	14	420
Rebre feedback final	12	CP	18	108

Refinaments	24	CP	18	432
Total				7812

Taula 5. Costos directes (Cas hipotètic)

3.1.3. Costos indirectes

Les despeses materials són molt poques. Això es deu al fet que quasi tot el material que s'utilitzarà durant el desenvolupament del projecte ja està adquirit. Com és el cas dels ordinadors: el de sobretaula, utilitzat en el lloc de treball, és un dels diversos ordinadors que disposen a l'empresa. El portàtil és un ordinador personal adquirit per motius acadèmics. Per tant, no s'ha cregut convenient incloure el preu d'ambdós ordinadors, ja que la despesa estava feta i no és afectada en absolut pel desenvolupament del projecte. De la mateixa manera tampoc s'ha cregut oportú incloure el lloguer (o el preu de compra) del lloc de treball ni de cap despesa compartida amb la resta de treballadors de l'empresa (com pot ser l'aire condicionat o la calefacció), ja que tampoc són despeses generades per aquest projecte.

El software també és un recurs important ha tenir en compte i que pot generar despeses. En el cas d'aquest projecte tot el software és d'ús gratuït o ja ha estat prèviament adquirit i per tant no genera cap despesa. El servidor on es desplegarà el sistema tampoc suposarà cap càrrec addicional, ja que actualment ja s'utilitza per a la resta del portal.

El que si s'ha tingut en compte és el consum elèctric propi del desenvolupador i el consum de material fungible. El consum elèctric del desenvolupador únicament és el generat per l'ordinador utilitzat per a desenvolupar el projecte. L'electricitat de les instal·lacions no es compta pel mateix motiu que s'ha esmentat abans, perquè són despeses compartides amb la resta de l'empresa. El material fungible és sobretot material d'oficina que es preveu utilitzar: una llibreta, *post-its* i bolígrafs.

Recurs	Quantitat	Cost	Total (85 dies)
Electricitat	2,1kWh/dia	0.145€/kWh	26€
Material fungible	---	---	10€
Total			36€

Taula 6. Costos indirectes

3.1.4. Cost total

El cost calculat fins ara és totalment estàtic, ja que només serà correcte si tot surt tal com s'ha plantejat. Per tant cal estudiar si els imprevistos poden augmentar el cost.

En cas que es necessiti més temps en el lloc de treball per acabar el desenvolupament del projecte el cost humà pot augmentar (si es necessita més temps fora del lloc de treball, el cost no augmenta, ja que no hi ha remuneració). Tenint en compte que la planificació es preveu totalment assolible la variació d'hores en cap cas hauria de ser superior al 10% del ja establert. Altrament significaria una mancança en la planificació o un imprevist molt important. Per tant la possible variació en el temps podria costar 476€ (10% de 4760€) o 771 (10€ de 7710) en el cas hipotètic plantejat en els costos directes.

En cas d'avaries del material el cost no augmentarà, ja que a l'empresa disposen d'ordinadors i de components de sobra. I en cas que l'ordinador personal patís algun desperfecte, l'empresa també ofereix portàtils si és necessari. Per tant no s'espera cap despesa addicional derivada dels materials.

Finalment, per cobrir possibles despeses que no s'hagin planificat, s'afegirà al cost màxim un percentatge de contingències per tal d'assegurar la fiabilitat del pressupost. S'ha estimat que el marge de contingències sigui del 5%.

Recursos	Cost real (€)	Cost hipotètic (€)
Costos directes	4760	7812
Costos indirectes	36	36
Desviacions	476	771
Contingències	263	425
Cost total del projecte	5535	9044

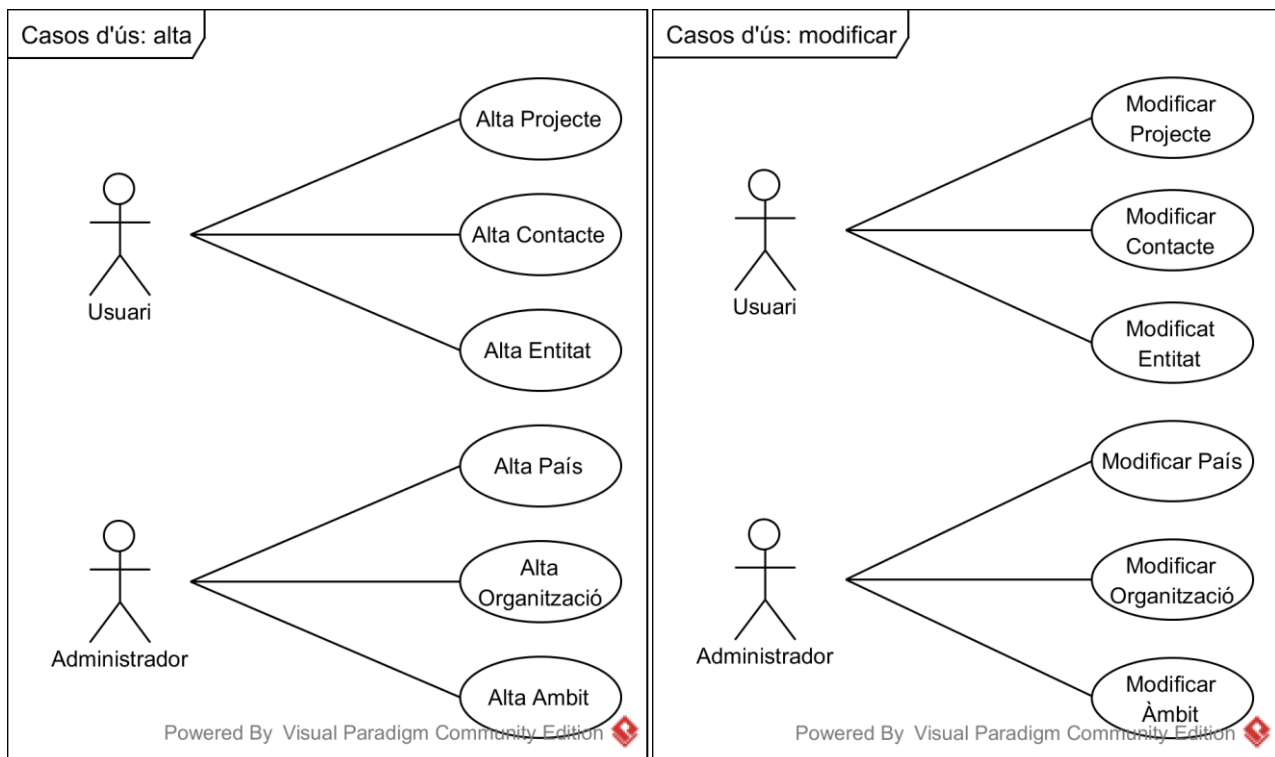
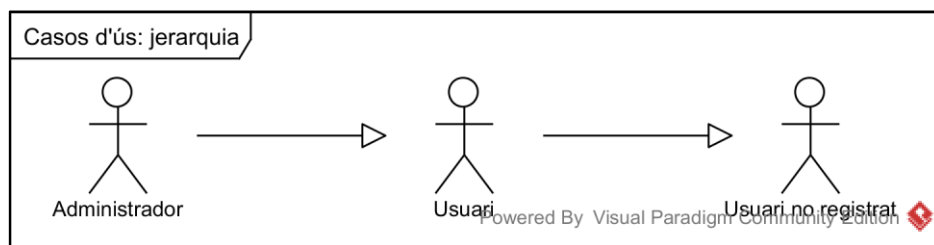
Taula 7. Costos totals

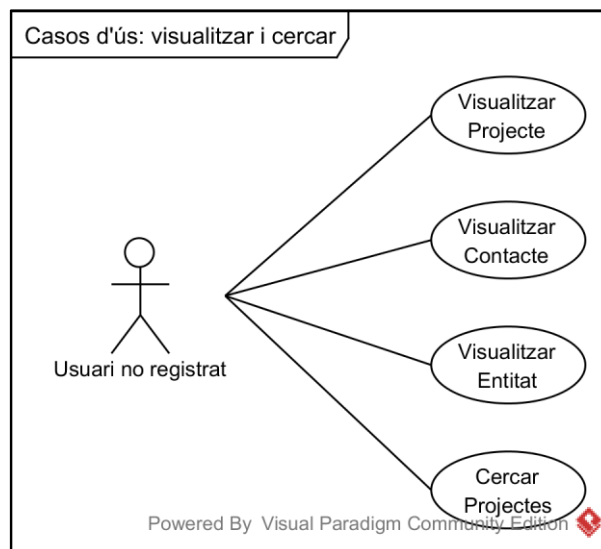
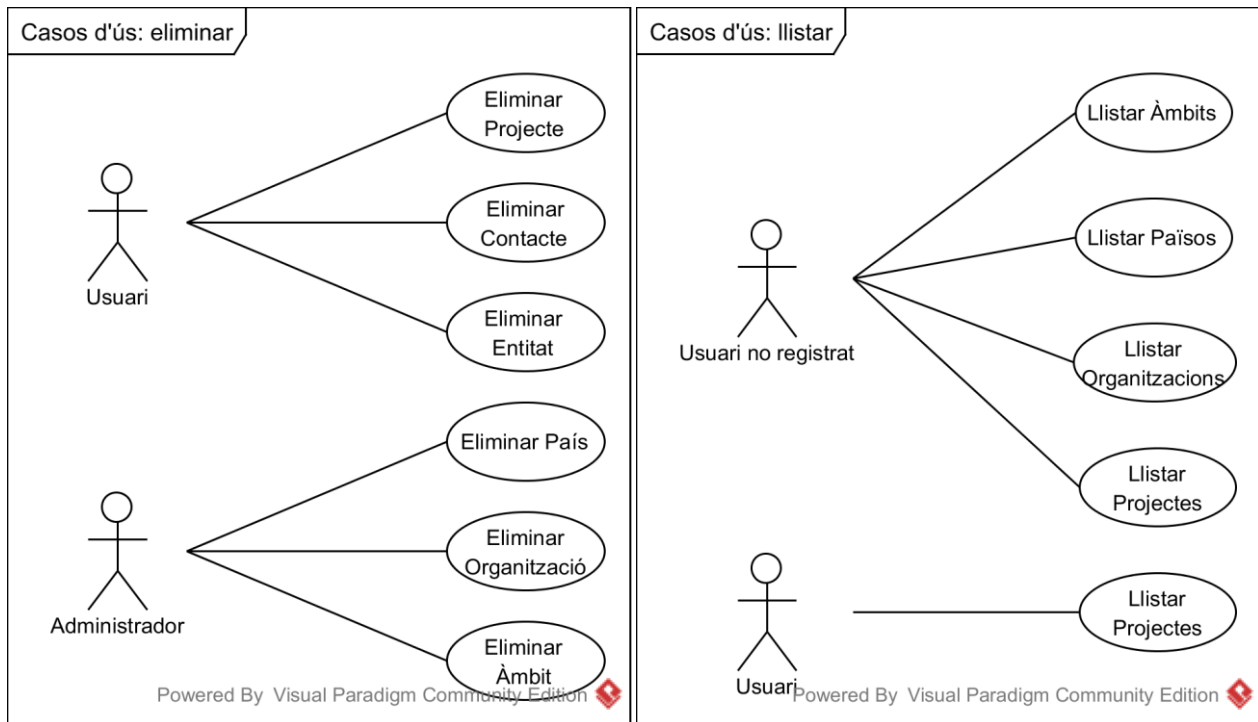
4. Especificació de requisits

En aquest capítol s'analitzaran els requisits del sistema. Aquests es dividiran en dos categories, els requisits funcionals i els no funcionals. Els requisits funcionals fan referència a funcionalitats que ha de disposar el sistema i estaran especificats en forma de casos d'ús. A més a més, podrem veure l'esquema conceptual de les dades. Els requisits no funcionals especifiquen les característiques o criteris de qualitat que ha de tenir el sistema. Estan especificats utilitzant les plantilles Volere.

4.1. Requisits funcionals

4.1.1. Diagrames de casos d'ús





4.1.2. Especificació casos d'ús

Cas d'Ús #01	Alta Projecte	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema donar d'alta un nou projecte		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona les dades del nou projecte Les dades obligatòries són: títol, data d'inici i data final Les dades opcionals són: acrònim, codi, fons de finançament, pressupost, pàgina web i imatge</div> <div>2. El sistema comprova les dades proporcionades</div> <div>3. El sistema crea un nou projecte amb les dades proporcionades i amb un id autogenerat</div>			
Extensions:			
<div>2A. Alguna de les dades obligatòries és buida o té un valor nul.<div>a. El sistema informa a l'usuari</div>b. El sistema finalitza el cas d'ús sense èxit</div> <div>2B. La data d'inici és major o igual a la data final<div>a. El sistema informa a l'usuari</div>b. El cas d'ús finalitza sense èxit</div>			

Cas d'Ús #02	Alta Contacte	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema donar d'alta un nou contacte		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona les dades del contacte Les dades obligatòries són: nom, adreça electrònica i adreça postal</div> <div>2. El sistema comprova les dades proporcionades</div> <div>3. El sistema crea un nou contacte amb les dades proporcionades i amb un id autogenerat</div>			
Extensions:			
<div>2A. Alguna de les dades obligatòries és buida o té un valor nul.</div> <div><div>a. El sistema informa a l'usuari</div><div>b. El sistema finalitza el cas d'ús sense èxit</div></div>			

Cas d'Ús #03	Alta Entitat	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema donar d'alta una nova entitat		
Escenari principal d'èxit:			
<div><div>1.</div><div>L'usuari proporciona les dades de la nova entitat Les dades obligatòries són: nom i país</div></div> <div><div>2.</div><div>El sistema comprova les dades proporcionades</div></div> <div><div>3.</div><div>El sistema crea una nova entitat amb les dades proporcionades i amb un id autogenerat</div></div>			
Extensions:			
<div><div>2A.</div><div>Alguna de les dades obligatòries és buida o té un valor nul.<div><div>a.</div><div>El sistema informa a l'usuari</div></div><div><div>b.</div><div>El sistema finalitza el cas d'ús sense èxit</div></div></div></div>			

Cas d'Ús #04	Alta País	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema donar d'alta un nou país		
Escenari principal d'èxit:			
<div><div>1.</div><div>L'administrador proporciona les dades del nou país Les dades obligatòries són: nom</div></div> <div><div>2.</div><div>El sistema comprova les dades proporcionades</div></div> <div><div>3.</div><div>El sistema crea un nou país amb les dades proporcionades</div></div>			
Extensions:			
<div><div>2A.</div><div>Alguna de les dades obligatòries és buida o té un valor nul.<div><div>a.</div><div>El sistema informa a l'usuari</div></div><div><div>b.</div><div>El sistema finalitza el cas d'ús sense èxit</div></div></div></div>			

Cas d'Ús #05	Alta Organització	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema donar d'alta una nova organització		
Escenari principal d'èxit:			
<div>1. L'administrador proporciona les dades de la nova organització Les dades obligatòries són: nom</div> <div>2. El sistema comprova les dades proporcionades</div> <div>3. El sistema crea una nova organització amb les dades proporcionades</div>			
Extensions:			
<div>2B. Alguna de les dades obligatòries és buida o té un valor nul.</div> <div><div>a. El sistema informa a l'usuari</div><div>b. El sistema finalitza el cas d'ús sense èxit</div></div>			

Cas d'Ús #06	Alta Àmbit	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema donar d'alta un nou àmbit		
Escenari principal d'èxit:			
<div><div>1.</div><div>L'administrador proporciona les dades del nou àmbit Les dades obligatòries són: nom</div></div> <div><div>2.</div><div>El sistema comprova les dades proporcionades</div></div> <div><div>3.</div><div>El sistema crea un nou àmbit amb les dades proporcionades i amb un id autogenerat</div></div>			
Extensions:			
<div><div>2A.</div><div>Alguna de les dades obligatòries és buida o té un valor nul.</div><div><div>a.</div><div>El sistema informa a l'usuari</div></div><div><div>b.</div><div>El sistema finalitza el cas d'ús sense èxit</div></div></div>			

Cas d'Ús #07	Modificar Projecte	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita modificar les dades d'un projecte existent		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona les dades a modificar. Les següents dades no poden ser buides al ser modificades: títol, data d'inici i data final La resta de les dades no tenen cap restricció al ser modificades: acrònim, codi, fons de finançament, pressupost, pàgina web i imatge</div> <div>2. El sistema comprova les dades proporcionades</div> <div>3. El sistema modifica les dades del projecte</div>			
Extensions:			
<div>2A. Alguna de les dades obligatòries és buida o té un valor nul.<div>a. El sistema informa a l'usuari</div><div>b. El sistema finalitza el cas d'ús sense èxit</div></div> <div>2B. La data d'inici és major o igual a la data final<div>a. El sistema informa a l'usuari</div><div>b. El cas d'ús finalitza sense èxit</div></div>			

Cas d'Ús #08	Modificar Contacte	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema modificar un contacte existent		
Escenari principal d'èxit:			
<div><div>1.</div><div>L'usuari proporciona les dades a modificar Les següents dades no poden ser buides al ser modificades: nom, adreça electrònica i adreça postal</div></div> <div><div>2.</div><div>El sistema comprova les dades proporcionades</div></div> <div><div>3.</div><div>El sistema modifica les dades del contacte</div></div>			
Extensions:			
<div><div>2A.</div><div>Alguna de les dades obligatòries és buida o té un valor nul.<div><div>a.</div><div>El sistema informa a l'usuari</div></div><div><div>b.</div><div>El sistema finalitza el cas d'ús sense èxit</div></div></div></div>			

Cas d'Ús #09	Modificar Entitat	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema modificar una entitat existent		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona les dades a modificar Les següents dades no poden ser buides al ser modificades: nom i país</div> <div>2. El sistema comprova les dades proporcionades</div> <div>3. El sistema modifica les dades de l'entitat</div>			
Extensions:			
<div>2A. Alguna de les dades obligatòries és buida o té un valor nul.</div> <div><div>a. El sistema informa a l'usuari</div><div>b. El sistema finalitza el cas d'ús sense èxit</div></div>			

Cas d'Ús #10	Modificar País	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema modificar un país existent		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona les dades a modificar Les següents dades no poden ser buides al ser modificades: nom</div> <div>2. El sistema comprova les dades modificades</div> <div>3. El sistema modifica les dades del país</div>			
Extensions:			
<div>2A. Alguna de les dades obligatòries és buida o té un valor nul.</div> <div><div>a. El sistema informa a l'usuari</div><div>b. El sistema finalitza el cas d'ús sense èxit</div></div>			

Cas d'Ús #11	Modificar Organització	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema modificar una organització existent		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona les dades a modificar Les següents dades no poden ser buides al ser modificades: nom</div> <div>2. El sistema comprova les dades modificades</div> <div>3. El sistema modifica les dades de l'organització</div>			
Extensions:			
<div>2B. Alguna de les dades obligatòries és buida o té un valor nul.</div> <div>c. El sistema informa a l'usuari</div> <div>d. El sistema finalitza el cas d'ús sense èxit</div>			

Cas d'Ús #12	Modificar Àmbit	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema modificar un àmbit existent		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona les dades a modificar Les següents dades no poden ser buides al ser modificades: nom</div> <div>2. El sistema comprova les dades modificades</div> <div>3. El sistema modifica les dades de l'àmbit</div>			
Extensions:			
<div>2A. Alguna de les dades obligatòries és buida o té un valor nul.</div> <div><div>e. El sistema informa a l'usuari</div><div>f. El sistema finalitza el cas d'ús sense èxit</div></div>			

Cas d'Ús #13	Eliminar Projecte	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema eliminar un projecte existent		
Escenari principal d'èxit:			
1. El sistema elimina el projecte			

Cas d'Ús #14	Eliminar Contacte	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema eliminar un contacte existent		
Escenari principal d'èxit:			
1. El sistema elimina el contacte			

Cas d'Ús #15	Eliminar Entitat	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema eliminar una entitat existent		
Escenari principal d'èxit:			
1. El sistema elimina l'entitat			

Cas d'Ús #16	Eliminar País	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema eliminar un país existent		
Escenari principal d'èxit:			
1. El sistema elimina el país			

Cas d'Ús #17	Eliminar Organització	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema eliminar una organització existent		
Escenari principal d'èxit:			
1. El sistema elimina l'organització			

Cas d'Ús #18	Eliminar Àmbit	Actor Principal:	Administrador
Disparador:	L'administrador sol·licita al sistema eliminar un àmbit existent		
Escenari principal d'èxit:			
1. El sistema elimina l'àmbit			

Cas d'Ús #19	Llistar Països	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema llistar tots els països existents		
Escenari principal d'èxit:			
1. El sistema llista els països existents, mostrant el nom de cadascun d'ells i ordenats alfabèticament			

Cas d'Ús #20	Llistar Organitzacions	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema llistar totes les organitzacions existents		
Escenari principal d'èxit:			
1. El sistema llista les organitzacions existents, mostrant el nom de cadascuna d'elles i ordenades alfabèticament			

Cas d'Ús #21	Llistar Àmbits	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema llistar tots els àmbits existents		
Escenari principal d'èxit:			
1. El sistema llista els àmbits existents, mostrant el nom de cadascun d'ells i ordenats alfabèticament			

Cas d'Ús #22	Llistar Entitats	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema llistar totes les entitats existents		
Escenari principal d'èxit:			
1. El sistema llista les entitats existents, mostrant el nom de cadascun d'elles i ordenades alfabèticament			

Cas d'Ús #23	Llistar Projectes	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema llistar tots els projectes existents		
Escenari principal d'èxit:			
1. El sistema llista els projectes existents, mostrant el nom i l'acrònim de cadascun d'ells i ordenats alfabèticament segons el nom			

Cas d'Ús #24	Llistar Projectes	Actor Principal:	Usuari
Disparador:	L'usuari sol·licita al sistema llistar tots els projectes del seu grup d'usuaris		
Escenari principal d'èxit:			
1. El sistema llista els projectes del grup de l'usuari, mostrant el nom i l'acrònim de cadascun d'ells i ordenats alfabèticament segons el nom			

Cas d'Ús #25	Visualitzar Projecte	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema visualitzar un projecte		
Escenari principal d'èxit:			
1. El sistema mostra totes les dades del projecte: títol, acrònim, codi, fons de finançament, data d'inici, data final, pressupost, pàgina web i imatge			

Cas d'Ús #26	Visualitzar Contacte	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema visualitzar un contacte		
Escenari principal d'èxit:			
1. El sistema mostra totes les dades del contacte: nom, adreça electrònica i adreça postal			

Cas d'Ús #27	Visualitzar Entitat	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema visualitzar una entitat		
Escenari principal d'èxit:			
1. El sistema mostra totes les dades de l'entitat: nom i país			

Cas d'Ús #28	Cercar Projectes	Actor Principal:	Usuari no registrat
Disparador:	L'usuari sol·licita al sistema cercar un projecte		
Escenari principal d'èxit:			
<div>1. L'usuari proporciona els paràmetres de la cerca:<div>a. Títol</div><div>b. Acrònim</div><div>c. Data: els projectes resultants de la cerca es trobaran actius, és a dir, que hauran</div></div>			

començat ($data \geq data_inici$) però encara no hauran finalitzat ($data \leq data_final$).

- d. Nom i/o país entitat: els projectes resultants de la cerca tindran associada una entitat (indiferentment si es amb el rol de *partner* o de coordinadora) amb el nom i/o el país proporcionats per l'actor.
2. El sistema cerca els projectes que compleixin els paràmetres de la cerca i els llista mostrant el nom i l'acrònim de cadascun d'ells i ordenats alfabèticament segons el nom

4.1.3. Model conceptual de les dades

El model conceptual de les dades és una percepció del món real que consisteix en una sèrie de classes relacionades entre elles. Les classes representen objectes o idees del món real. Per representar les característiques d'aquestes classes s'utilitzen els atributs. Finalment les classes poden dependre unes de les altres, sent representada aquesta dependència per les relacions entre elles.

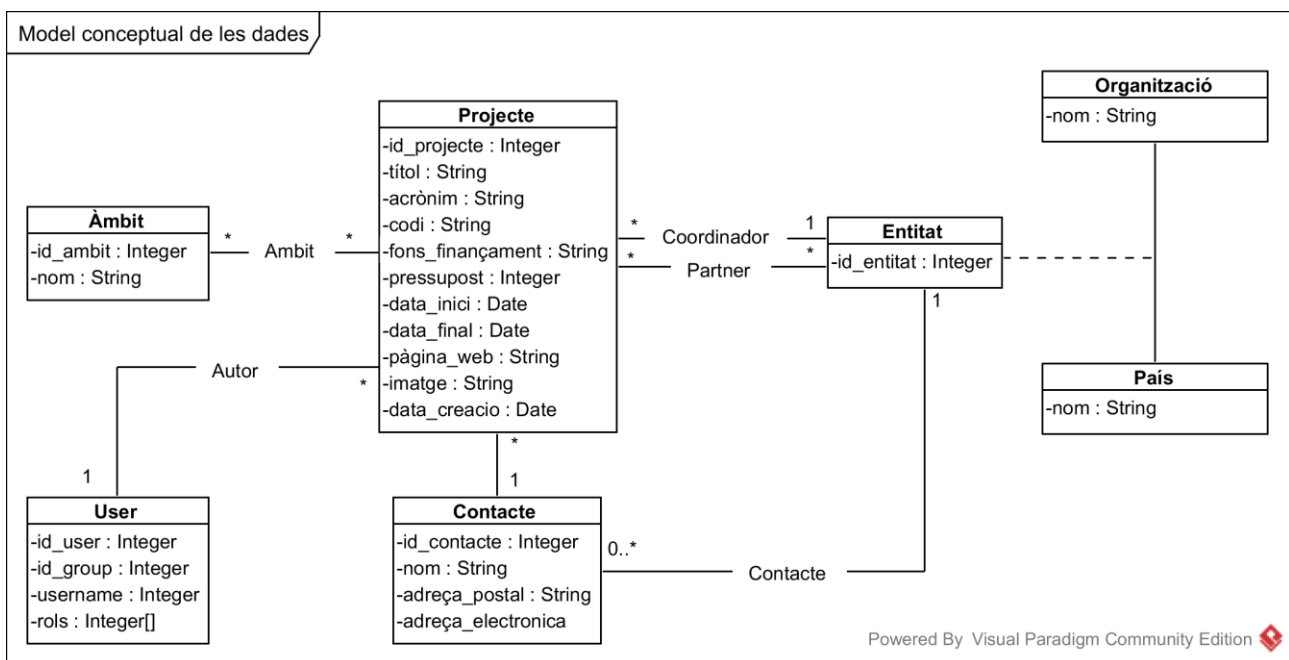


Figura 5. Model conceptual de les dades

Restriccions textuais

- Claus externes: Àmbit (id_ambit), Contacte (id_contacte), Entitat (id_entitat), Organització(nom), País(nom), Projecte(id_projecte), User (id_usuari).
- Una entitat coordinadora no pot ser *partner* en el mateix projecte.
- L'entitat del contacte ha de ser coordinadora del projecte al qual esta associat.
- En un projecte, la data_inici no pot ser més gran que la data_final (data_inici =< data_final).

4.1.4. Descripció de les classes

Àmbit

Classe que representa un tema o tòpic el qual pot estar relacionat amb un projecte.

Atribut	Estructura	Descripció
id_ambit	Integer	Identificador de l'atribut
nom	String	Nom de l'àmbit

Contacte

Classe que representa una persona amb la qual es pot contactar per assumptes relacionats amb el projecte al qual està associada.

Atribut	Estructura	Descripció
id_contacte	Integer	Identificador del contacte
nom	String	Nom complet del contacte
adreça_electrònica	String	Adreça electrònica del contacte
adreça_postal	String	Adreça postal on es troba el contacte en horari laboral

Entitat

Classe que representa una organització o empresa en un país concret, sent el país el lloc principal on és realitza el projecte al qual està associada.

Atribut	Estructura	Descripció
id_entitat	Integer	Identificador de l'entitat

Organització

Classe que representa una organització o empresa, la qual pot ser potencialment participant o coordinadora d'un projecte.

Atribut	Estructura	Descripció
nom	String	Nom de l'organització

País

Classe que representa un país, el el qual es pot dur a terme potencialment un o més projectes.

Atribut	Estructura	Descripció
nom	String	Nom del país

Projecte

Classe que representa un projecte, el qual sempre està associat a una entitat amb el rol de coordinadora, amb un contacte i amb un usuari (l'usuari que ha creat el projecte)

Atribut	Estructura	Descripció
id_projecte	Integer	Identificador del projecte
títol	String	Títol del projecte
acrònim	String	Acrònim del projecte
codi	String	Codi del projecte. És únic
fons_finançament	String	Fons de finançament del projecte
pressupost	Integer	Pressupost total del projecte, en euros.

data_inici	Date	Data inicial del projecte
data_final	Date	Data de finalització del projecte
pàgina web	String	URL de la pàgina web del projecte
logo	String	URL del logotip del projecte
data_creacio	Data	Data de la creació del projecte en el sistema

User

Classe que representa un usuari del portal RuralCat. Aquesta classe és compartida amb la resta del portal i conté més atributs dels mostrats en el model conceptual, només es mostren els importants per al nou sistema que es vol construir.

Atribut	Estructura	Descripció
id_usuari	Integer	Identificador de l'usuari
id_grup	Integer	Identificador del grup d'usuaris al qual pertany l'usuari
username	String	No del usuari en el sistema
rols	Integer[]	Identificadors dels rols associats a l'usuari

4.2. Requisits no funcionals

Requisit #:	2 i 4	Tipus de requisit segons Volere:	11a. Easy of Use 11c. Learning
Descripció:	El sistema serà fàcil i intuïtiu d'utilitzar		
Justificació:	Un dels majors objectius del sistema es facilitar la gestió dels projectes, per tant, si el sistema no es fàcil d'utilitzar no podrem assolir aquest objectiu principal.		
Condicció de satisfacció:	Un 80% els usuaris podran utilitzar el sistema sense cap mena d'instrucció, de forma totalment autònoma.		

Requisit #:	5	Tipus de requisit segons Volere:	11d.Undertandability and Politeness
Descripció:	Les explicacions del producte no poden contenir ambigüitats.		
Justificació:	L'usuari ha de ser capaç d'entendre el que el sistema li vol comunicar.		
Condicció de satisfacció:	El 100% del usuaris afirmen que els missatges del sistema són totalment explicatius i no contenen ambigüitats.		

Requisit #:	7	Tipus de requisit segons Volere:	12a. Speed and Latency
Descripció:	La interfície pública (la que contindrà la funcionalitat de cercar projectes) tindrà un temps de resposta màxim de 2 segons, la resta 5 segons.		

Justificació:	No volem que les usuaris deixin de cercar projectes per que el sistema els resulti lent d'utilitzar, tot i això no es tant preocupant que el sistema pugui tardar una mica (sense superar un marge raonable) en la part privada d'administració.
Condicció de satisfacció:	El producte ha de respondre en un temps inferior a 2 segons pel 90% de les sol·licituds en la part pública. Cap resposta ha de trigar més de 5 segons.

Requisit #:	8	Tipus de requisit segons Volere:	12d. Reliability and Availability
Descripció:	El sistema ha d'estar disponible 24 hores al dia, 365 dies a l'any		
Justificació:	El client vol disponibilitat total del sistema durant tot l'any		
Condicció de satisfacció:	El sistema estarà disponible sempre.		

Requisit #:	11	Tipus de requisit segons Volere:	12g. Scalability or Extensibility
Descripció:	El sistema està preparat per incorporar funcionalitats similars o afegir-ne de noves.		
Justificació:	Auto-explicatiu.		
Condicció de satisfacció:	S'aplicaran patrons de disseny (MVC i Interfície) que permetin implementar noves funcionalitats en el futur sense modificar les actuals.		

Requisit #:	14	Tipus de requisit segons Volere:	14a. Maintenance
Descripció:	Tots les funcionalitats del sistema podran ser modificades fàcilment per adaptar-se a noves necessitats		
Justificació:	S'han de poder modificar tots els aspectes que puguin variar després de la realització del projecte.		
Condicció de satisfacció:	Les funcions i les variables tindran noms explicatius, si no és possible, es documentarà degudament amb comentaris. A més a més el codi haurà el més estàndard possible per tal de ser comprensible fàcilment.		

Requisit #:	15	Tipus de requisit segons Volere:	14b. Support
Descripció:	Els usuaris podran contactar amb un equip de suport		
Justificació:	Volem ser capaços d'ajudar amb qualsevol tipus de problema que puguin tenir el usuaris.		
Condicció de satisfacció:	Els usuaris podran contactar via mail per resoldre qualsevol dubte que puguin tenir sobre el sistema		

Requisit #:	16	Tipus de requisit segons Volere:	14c. Adaptability
Descripció:	S'espera que el producte es pugui executar en els navegadors web més utilitzats		
Justificació:	Volem que els usuaris no tinguin restriccions a la hora d'utilitzar el sistema. S'ha de tenir en compte que el target pot tenir més tendència que altres col·lectius a utilitzar versions de navegadors antigues.		

Condició de satisfacció:	El sistema serà compatible almenys amb Chrome, Firefox, Safari, Opera i Internet explorer (en totes les seves versions)
---------------------------------	---

Requisit #:	17	Tipus de requisit segons Volere:	15a. Access
Descripció:	Els usuaris que no tinguin permisos d'administrador en cap cas podran donar d'alta ni modificar cap projecte ni la informació associada a aquests.		
Justificació:	Els usuaris amb drets d'administració volen poder controlar els seus projectes.		
Condició de satisfacció:	Les funcionalitats d'alta i de modificació seran exclusives dels usuaris registrats amb el rol d'administrador		

Requisit #:	18	Tipus de requisit segons Volere:	15b. Integrity
Descripció:	El sistema no acceptarà ni guardarà dades que puguin ser incorrectes.		
Justificació:	És molt important mantenir la integritat de les dades al sistema per tal d'assegurar la seva correctesa i validesa.		
Condició de satisfacció:	El sistema no permet violar cap restricció d'integritat definida explícitament.		

5. Arquitectura

Aquest capítol es centrarà en el disseny del sistema. En primer lloc es definirà i justificarà l'arquitectura global del sistema. En segon lloc podrem veure el diagrama de classes de disseny, una refinació del model conceptual de les dades. S'enumeraran i descriuran els patrons de disseny utilitzats en la implementació del sistema. També es podran veure els diagrames de seqüència de les funcionalitats principals del sistema i les interfícies gràfiques. Finalment veurem com està organitzada la base de dades, des del punt de vista lògic i físic.

5.1. Visió global

Un dels aspectes més importants d'un projecte software és l'arquitectura utilitzada. En aquest projecte s'utilitzarà una arquitectura clàssica de tres capes: presentació, domini i persistència (vegeu figura 6).

La capa de presentació contindrà totes les vistes i els seus controladors. Per tant, està destinada exclusivament a la part visual del sistema. La capa del domini s'encarrega de gestionar els models i la interacció entre ells. A més a més, proporciona les dades a mostrar a la capa de presentació i es comunica amb la capa de persistència per obtenir, actualitzar, donar d'alta i eliminar les dades emmagatzemades a la BBDD. Per tant, és la capa que engloba tota la lògica i la més important del sistema. Finalment la capa de persistència engloba tota la comunicació amb la BBDD.

El principal avantatge d'aquesta arquitectura és que cada capa és independent de la resta, actuant cadascuna com un bloc diferenciat. Això ens permet substituir un bloc, com per exemple la capa de persistència, per un d'altre i el sistema podria seguir funcionant (sempre que es respectin els noms de les crides a les funcions i els seus paràmetres).

5.2. Diagrama de classes de disseny

Per realitzar el diagrama de classes de disseny s'ha partit de l'esquema conceptual de les dades (capítol 4.1.3). En primer lloc s'ha eliminat la classe associativa, encapsulant tota la informació en una sola classe. És el cas de la classe *Entitat*, la qual ha agrupat la classe associativa amb el mateix nom, la classe *Organització* i la classe *País*. En segon lloc s'han definit les navegabilitats, és a dir, s'ha especificat quina classe té visibilitat sobre l'altre. La classe que tingui visibilitat sobre un altre, a part dels seus propis atributs, tindrà un pseudoatribut que apunta a la instància amb la qual es relaciona, fent possible la navegació entre classes. Per exemple, en el cas d'aquest projecte, podem veure al diagrama (vegeu figura 7) que la classe *Projecte* té visibilitat sobre *Entitat* amb dos rols diferents. Per tant, tota instància de *Projecte* tindrà dos pseudoatributs: *entitatCordinadora*, el qual farà referència a una instància d'*Entitat*, i *partners*, el qual serà una llista d'*Entitats*.

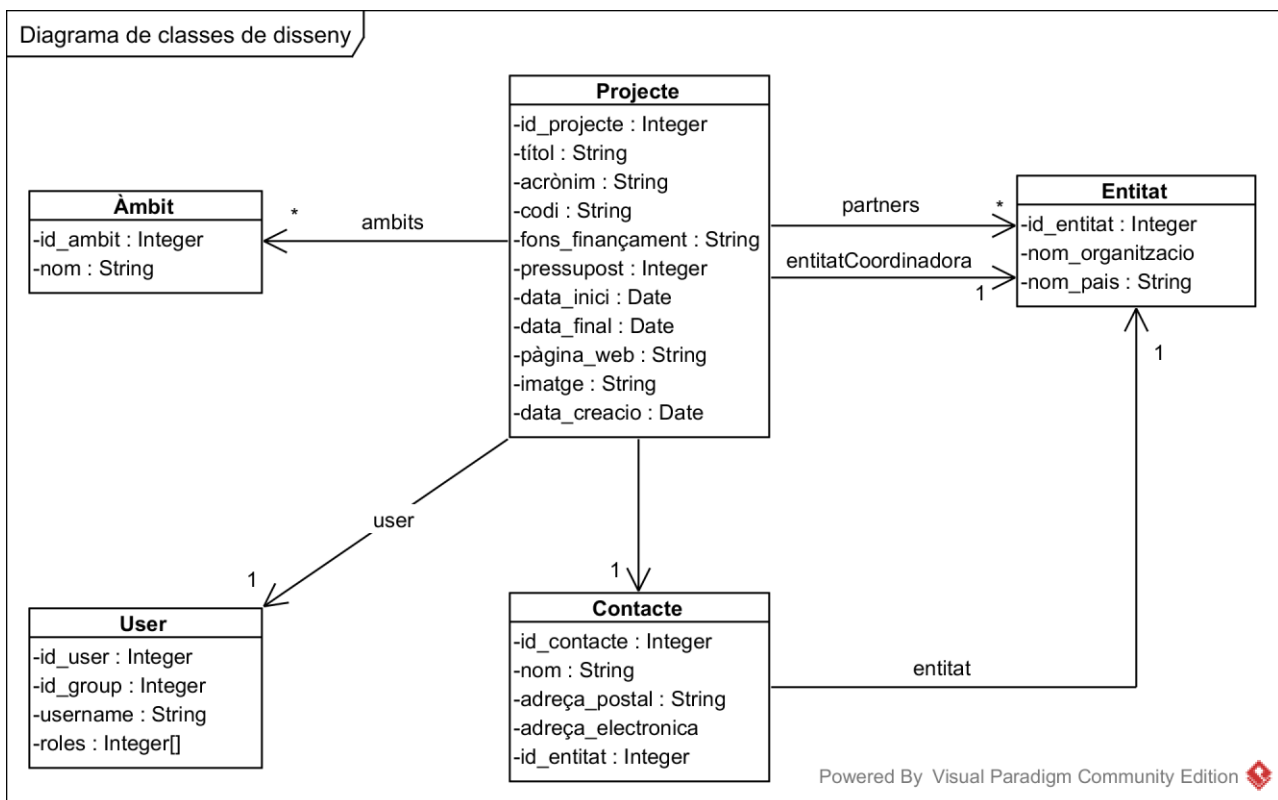


Figura 6. Diagrama de classes de disseny

5.3. Patrons de disseny

Els patrons de disseny són solucions genèriques a problemes comuns i recurrents en el disseny del software. No són dissenys totalment definits, sinó que actual com una plantilla per resoldre el problema al qual s'adrecen, sent possible que siguin utilitzats en situacions molt diverses i en diversos llenguatges de programació. La principal avantatge d'utilitzar aquests patrons és que ofereixen solucions provades i eficients, en molts casos sent l'opció més viable. A més a més, al ser solucions conegudes per a tots els desenvolupadors, és més fàcil entendre un disseny que utilitzi un d'aquests patrons.

A continuació es citaran els patrons de disseny utilitzats en aquest projecte, així com una descripció de quin problema busquen resoldre i quina solució ofereixen:

Patró MVC

Aquest patró de disseny es utilitza per implementar interfícies d'usuari. Consisteix en dividir l'aplicació en tres parts interconnectades (vegeu figura 8):

- **Model:** s'encarrega de gestionar la lògica del sistema, és a dir, la interacció amb les classes del domini i amb la capa de persistència per realitzar les crides a la BBDD.
- **Vista:** és la part encarregada de mostrar d'interfície a l'usuari, mostra les dades però no té cap responsabilitat sobre elles.
- **Controlador:** és la capa intermediària entre el model i la vista. S'encarrega de gestionar les interaccions de l'usuari amb la vista (invocant peticions al model) i de gestionar quina vista es mostra en cada moment.

Les principals avantatges d'utilitzar aquest patró és que ens permet separar els components del nostre sistema en funció de la seva responsabilitat (responsabilitat única), fent molt més senzilla la tasca de realitzar canvis en un component concret. Per exemple, podríem canviar la implementació del model sense que la vista i el controlador patissin cap canvi, això afavoreix un manteniment eficient del sistema, resultant en un estalvi d'hores de treball a llarga durada.

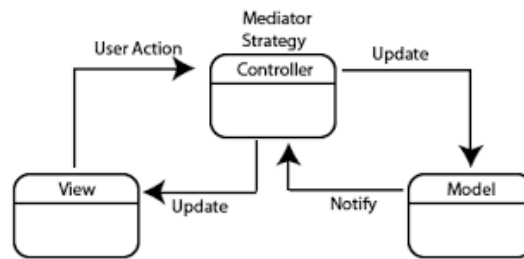


Figura 7. Patró MVC

Patró Singleton

És un patró de disseny que s'utilitza per restringir el número d'instàncies d'una classe a un sol objecte que és accessible de forma global. És útil quan solament es necessita un objecte per coordinar les accions dins el sistema. Consisteix en crear un mètode en la classe que esdevindrà singleton que crea una instància del objecte si no existeix cap. Per assegurar-se que solament es crea una instància es regula el constructor.

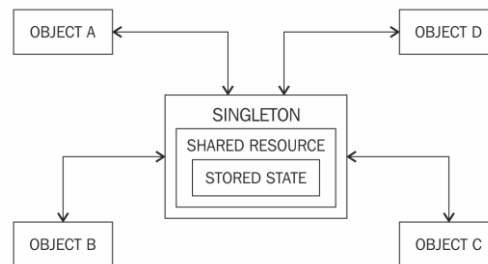


Figura 8. Patró Singleton

Patró interfície

La idea principal d'una interfície és separar la signatura d'una funció de la seva implementació. Per tant les interfícies només engloben signatures de funcions que hauran de ser implementades per les classes que implementin la interfície.

La principal avantatge d'aquest patró és que abstraïu la resta del sistema de la implementació de la qual es fa responsable la interfície, sent fàcilment intercanviable, ja que els mètodes ja estan definits. En aquest projecte s'utilitza per abstraure les diferents capes de l'arquitectura.

5.4. Exemples de diagrames de seqüència

Els diagrames de seqüència mostren com es comporta el sistema, és a dir, com interactuen els diferents objectes entre si i amb l'actor que desencadena el cas (o casos) d'ús. Cada objecte és representat amb una caixa amb el seu nom a dins i amb una línia vertical anomenada línia de vida. Les diferents interaccions es representen unint les línies de vida amb una fletxa amb el concepte de la interacció. Moltes d'aquestes interaccions representen funcions del sistema i per tant apareixen amb la seva signatura (nom i paràmetres de la funció). Tot i això, en el cas de les funcions que tenen paràmetres, s'han obviat en el diagrama (substituint-los per (...)) per tal de millorar la facilitat d'enteniment de l'esquema, ja que en molts casos el número de paràmetres era força elevat. Igualment, els paràmetres s'especificaran com a suport del diagrama.

Alta Projecte

Aquest diagrama (figura 10) mostra les accions que realitza el sistema per donar d'alta un projecte. Tot i que el cas d'ús principal al qual dona resposta és el cas d'ús #01 *Alta Projecte*, també inclou molts d'altres, com per exemple, #03 *Alta Entitat*. Aquesta acció només la poden realitzar els usuaris amb el rol d'usuari registrat (que inclou el rol d'administrador i exclou el d'usuari no registrat)

El diagrama comença amb la interacció de l'usuari sobre el sistema, a l'indicar que vol guardar un nou projecte. Es comprova que totes les dades de caràcter obligatori hagin estat introduïdes, en cas que no es compleixi aquesta condició, el sistema retornarà un missatge a l'usuari especificant quines dades falten per ser introduïdes. Un cop totes les dades siguin correctes el sistema procedirà a realitzar l'alta del projecte.

En les operacions SQL que es realitzen (invocades per ProjectesDAO) es comprova si ja existeix un objecte amb els mateixos atributs. En cas afirmatiu es retorna l'objecte ja existent, en cas negatiu es crea i es retorna el nou objecte.

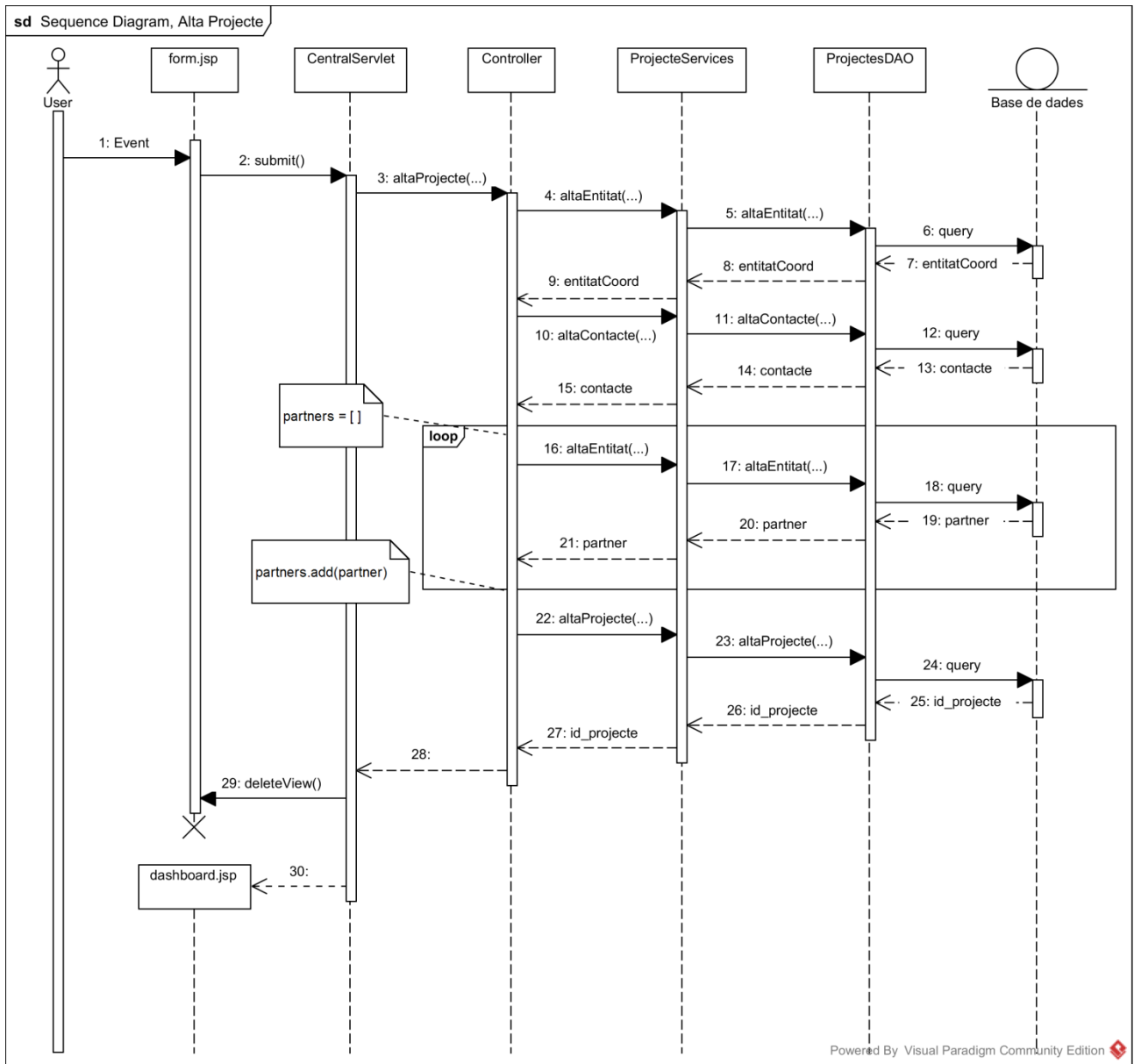


Figura 9. Diagrama de seqüència: alta projecte

(3) altaProjecte (String títol, String acrònim, String codi, String fons_finançament, Integer pressupost, Date data_inici, Date data_final, String pàgina_web, String imatge, String nom_entitat_coordinadora, String pais_entitat_coordinadora, List<String> nom_entitats_partner, List<String> països_entitats_partner, String nom_contacte, String adreça_postal, String adreça_electronica, List<String> nom_ambits, Integer id_user, Integer id_group, Date data_creacio)

(4,5,16 i 17) altaEntitat (String nom_entitat_coordinadora, String pais_entitat_coordinadora)

(10 i 11) altaContacte (String nom_contacte, String adreça_postal, String adreça_electronica, Entitat entitatCoord)

(22 i 23) altaProjecte (String títol, String acrònim, String codi, String fons_finançament, Integer pressupost, Date data_inici, Date data_final, String pàgina_web, String imatge, List<String> nom_ambits, Integer id_user, Integer id_group, Date data_creacio, Entitat entitatCoord, Contacte contacte)

Cercar Projecte

En aquest diagrama (figura 11) podem veure com es comporta el sistema per cercar projectes. Aquesta acció correspon principalment al cas d'ús #28 Cercar projectes, però a l'igual que en el cas anterior també inclou altres casos d'ús. L'actor és un usuari amb el rol d'usuari no registrat (guest per abreviar).

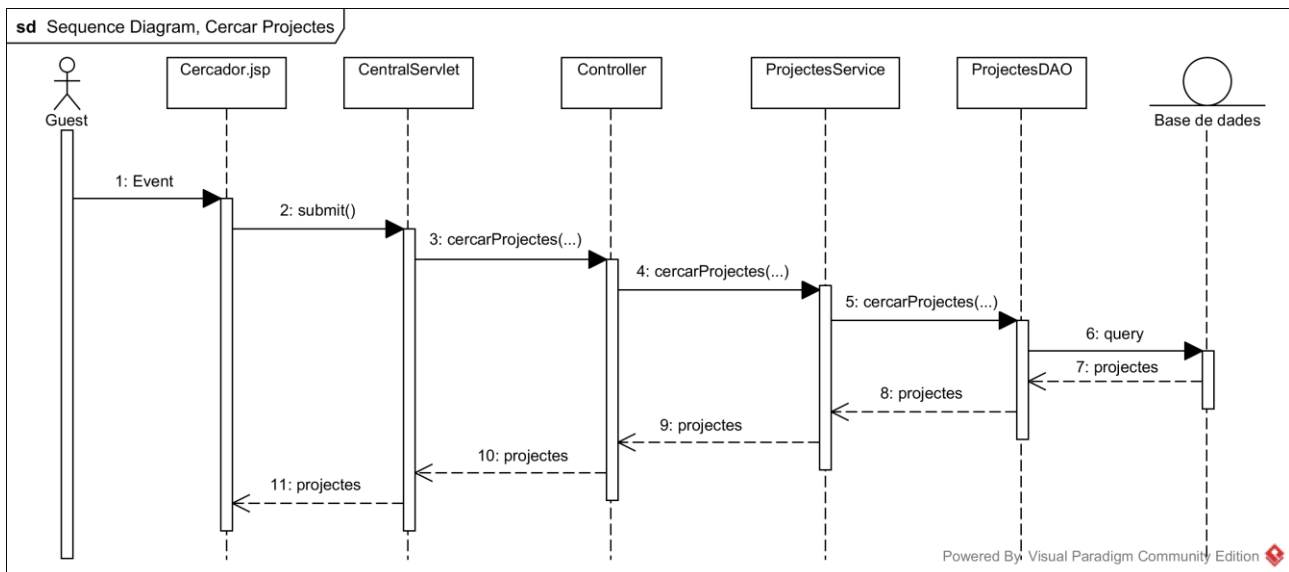


Figura 10. Diagrama de seqüència: cercar Projecte

(3, 4 i 5) cercarProjectes (String títol, String acrònim, Integer any, String nom_entitat, String pais_entitat, List<String> nom_ambits)

5.5. Disseny de la interfície

En aquest apartat es mostraran les principals interfícies del sistema. Com es podrà observar totes tenen un estil similar, per tal d'aconseguir una bona coherència estètica dins el sistema. A més a més els colors utilitzats són els de Ruralcat i el disseny del cercador s'adequa al de la resta de cercadors existents al portal.

The screenshot displays a search interface with a sidebar on the left containing various filters and a main area on the right showing search results. The sidebar filters include:

- Cerca per títol:** A search input field with a magnifying glass icon.
- Filtra per acrònim:** An input field for filtering by acronym.
- Filtra per any de vigència:** A dropdown menu labeled "Selecciona un any" with a calendar icon.
- Filtra per fons de finançament:** A dropdown menu labeled "Selecciona un fons de finançament".
- Filtra per entitat:** A dropdown menu labeled "Selecciona una entitat".
- Filtra per país de l'entitat:** A dropdown menu labeled "Selecciona un país de l'entitat".
- Filtra per àmbits:** A list of categories including Agricultura de precisió, Apicultura, Arròs, Avicultura, Cadena i economia agroalimentària, Cereals d'estiu, Cereals d'hivern, and Citricultura.

At the bottom of the sidebar are buttons for "Cercar" (Search) and "Netejar" (Clear).

The search results on the right are displayed in a list format, each with a title, funding source, coordinator entity, validity period, and country, followed by a "Visualitza" (View) button:

- Aprovechamiento Integral de Biomasa Agroalimentaria**
Fons de finançament: MINECO - Plan Nacional I+D+I | Vigència: 2016-2018
Entitat coordinadora: Universitat de Lleida | País de l'entitat: Espanya
- Implantació de l'entorn natural als espais urbans**
Fons de finançament: AGL2016-76943-C2-1-R | Vigència: 2015-2018
Entitat coordinadora: Universitat de Lleida | País de l'entitat: Espanya
- Filtres granulars en sistemes de reg per degoteig amb aigües residuals regenerades: condicions operatives i consum energètic (FILGRAGOTAR)**
Fons de finançament: MINECO - Plan Nacional I+D+I | Vigència: 2009-2011
Entitat coordinadora: Universitat de Barcelona | País de l'entitat: Espanya
- Development and application of Novel, Integrated Tools for monitoring and managing Catchments (INTCATCH)**
Fons de finançament: HORIZON 2020 | Vigència: 2016-2019
Entitat coordinadora: Brunel University London | País de l'entitat: Regne Unit
- Diseño de sistemas nanoestructurados para proteger y liberar compuestos naturales con actividad funcional y tecnológica (NANOPROTECT)**
Fons de finançament: HORIZON 2020 | Vigència: 2014-2018
Entitat coordinadora: Universitat de Lleida | País de l'entitat: Espanya

At the bottom of the results area are navigation controls: "Pàgina 1 de 2", "Articles per pàgina", and buttons for "← Primer", "Anterior", "Següent", and "Últim →".

Figura 11. Interfície cercador

INFORMACIÓ DEL PROJECTE

ACRÒNIM	
FILGRAGOTAR	
TÍTOL	
Filtres granulars en sistemes de reg per degoteig amb aigües residuals regenerades: condicions operatives i consum energètic	
PRESSUPOST TOTAL	
75000 €	
FONS DE FINANÇAMENT	
MINECO - Plan Nacional I+D+i	
DATA d'INICI	DATA de FINALITZACIÓ
Febrer 2018	Març 2020
DESCRIPCIÓ DEL PROJECTE	
El projecte vol determinar com afecta l'alçada del llit filtrant i la velocitat de filtració en la qualitat de l'aigua filtrada, en l'obtenció dels degoters, en la caiguda de pressió en els filtres i en el consum energètic dels sistemes de reg per degoteig. Per això, s'assajaran 3 tipus de filtres granulars diferents que s'emplenaran amb 3 materials diferents.	
ENTITAT RESPONSABLE (CAP DE FILES) DEL PROJECTE	
Universitat de Barcelona	Espanya
ENTITATS PARTICIPANTS DEL PROJECTE	
Universitat de Girona	Espanya
Universitat de Lleida	Espanya
CONTACTE DEL PROJECTE A CATALUNYA	
NOM:	Jaume Puig Blasco
E-MAIL:	jpuig@ub.edu
NOM ENTITAT:	Universitat de Barcelona
PÀGINA WEB	LOGOTIP
http://webdelprojecte.cat	http://urldelaimatge.com
ÀMBIT/S DEL PROJECTE	
<div>Cadena i economia agroalimentària</div> <div>Cereals d'hivern</div> <div>Citricultura</div>	

Figura 12. Interfície visualitzar projecte

PROJECTE			
Títol*			
<input type="text"/>			
Acrònim		Codi	
<input type="text"/>		<input type="text"/>	
Fons de finançament*		Pressupost total	
<input type="text"/>		<input type="text"/>	
Mes d'inici*	Any d'inici*	Mes de finalització*	Any de finalització*
Selecciona un mes ▼	Selecciona un any ▼	Selecciona un mes ▼	Selecciona un any ▼
Pàgina web		Logo	
<input type="text"/>		<input type="button" value="Examinar..."/> No se ha seleccionado ningún archivo.	
Descripció			
<input type="text"/>			
ENTITAT COORDINADORA DEL PROJECTE			
Nom*		País*	
Selecciona una entitat ▼		Selecciona un país ▼	
ENTITATS PARTICIPANTS DEL PROJECTE			
<input type="button" value="Afegir partner"/>			
CONTACTE DEL PROJECTE A CATALUNYA			
Nom*	Primer Cognom*	Segon Cognom	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Adreça postal*		E-mail*	
<input type="text"/>		<input type="text"/>	
ÀMBIT/S DEL PROJECTE			
<div> Agricultura de precisió Apicultura Arròs Avicultura Cadena i economia agroalimentària Cereals d'estiu Cereals d'hivern Citricultura Cunicultura Estalvi energètic i energies renovables Farratges, proteaginoses i oleaginoses Fruita dolça Fruita seca </div>			
<input type="button" value="Tornar"/> <input type="button" value="Guardar"/>			

Figura 13. Interfície crear/editar projecte

Administració Projectes Europeus

Group 2

Acrònim	Títol del projecte	Accions
	Aprovechamiento Integral de Biomasa Agroalimentaria	editar eliminar
INTCATCH	Development and application of Novel, Integrated Tools for monitoring and managing Catchments	editar eliminar
NANOPROTECT	Diseño de sistemas nanoestructurados para proteger y liberar compuestos naturales con actividad funcional y tecnológica	editar eliminar
FILGRAGOTAR	Filtres granulars en sistemes de reg per degoteig amb aigües residuals regenerades: condicions operatives i consum energètic	editar eliminar
	Implantació de l'entorn natural als espais urbans	editar eliminar
POSTFIRE	Mortalidad arbórea tras incendios: procesos subyacentes y consecuencias para la recuperación y gestión de los bosques mediterráneos	editar eliminar
RISALTE	Rivers of salt: advancing towards a better management of salt-rich effluents	editar eliminar
7 projectes		

[Crear Projecte](#)

Figura 14. Interficie dashboard

5.6. Bases de dades

Durant l'execució d'un sistema de software, aquest pot mantenir, de forma temporal, les dades modificades o noves. Però, un cop el sistema finalitza la seva execució, si no emmagatzema aquestes dades, es perdran. És per aquest motiu que gairebé tot projecte de software necessita una base de dades on guardar els canvis en el sistema, és a dir, un conjunt de dades segons una estructura coherent on qualsevol d'aquestes dades pot ser extreta del conjunt i actualitzada, sense afectar ni l'estructura del conjunt ni les altres dades.

En aquest projecte s'utilitza una base de dades relacional, model en el qual les dades es guarden en diferents *tuples* i es relacionen entre elles, i el llenguatge SQL per realitzar les diferents operacions sobre les dades. Per gestionar la base de dades s'ha utilitzat el gestor de bases de dades relacionals (RDBMS) MySQL.

En aquest capítol es mostra els dissenys que s'han utilitzat en aquest projecte: tant el disseny lògic, que mostra com s'organitzen i es relacionen les dades, com el disseny físic, que mostra com realment es guarden les dades.

5.6.1. Disseny lògic

El disseny lògic mostra quines dades es guardaran a la base de dades i com es relacionaran entre elles. És un diagrama molt similar al del model conceptual de les dades, amb l'excepció que la classe associativa *Entitat* entre *Organització* i *País* s'ha substituït per una única classe entitat amb els atributs de les classes anteriors.

En aquest esquema no s'especifica cap clau primària ni cap clau forana, ja que, tot i que totes les dades que es mostren es guardaran a la base de dades, quan es realitzi el disseny físic les taules (conjunts de tuples amb un mateix esquema) es poden veure modificades o poden necessitar-se de noves (per suportar associacions *many-to-many* per exemple).

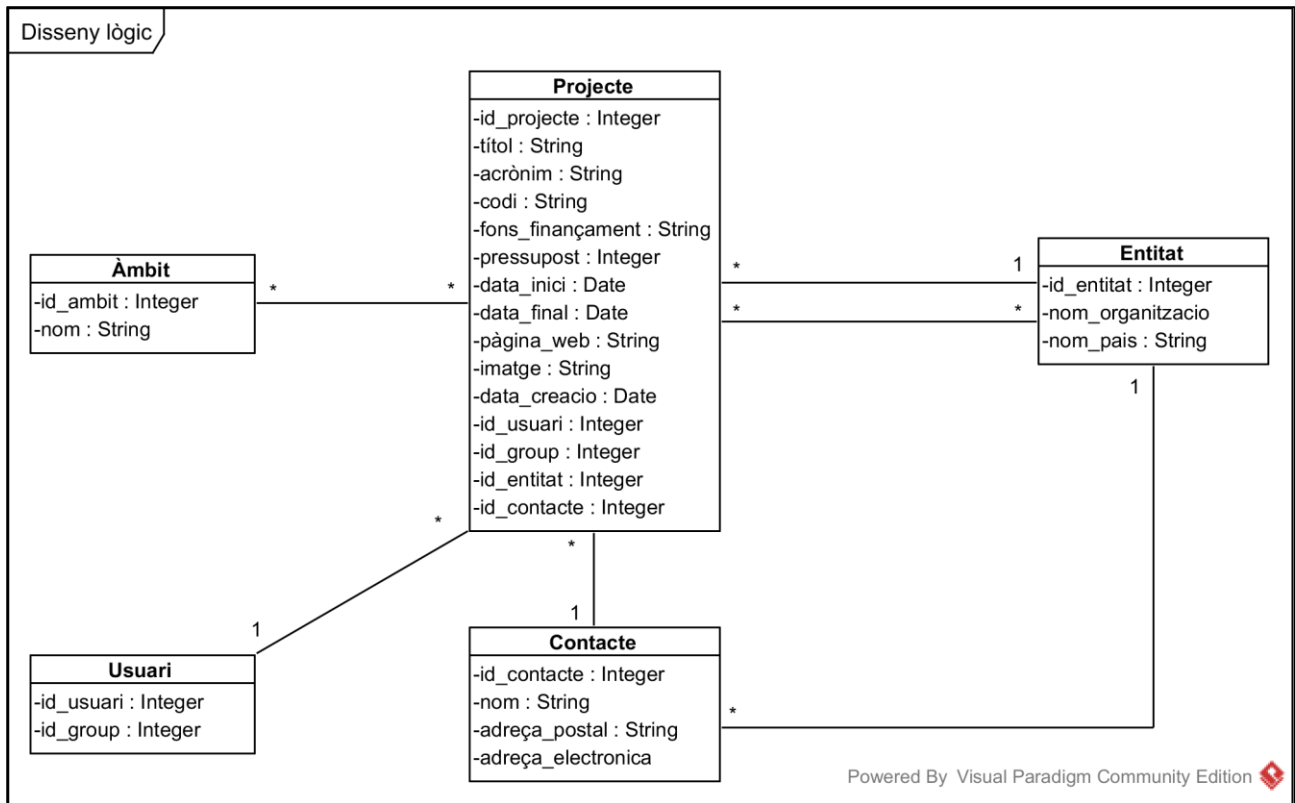


Figura 15. Disseny lògic

5.6.2. Disseny físic

El disseny físic mostra com es guardaran les dades a la base de dades. Per realitzar aquest diagrama partim del disseny lògic i per tant cal explicar quines transformacions s'han realitzat.

En primer lloc s'han creat dues noves taules: *ProjectesAmbits* i *ProjectesEntitats*. Aquestes noves taules serveixen per mantenir la relació *many-to-many* entre *Projectes-Ambits* i entre *Projectes-Entitats* (amb el rol de *partner*) respectivament i guarden únicament les claus primàries de les dues *tuples* que relacionen.

En segon lloc per mantenir l'associació *one-to-many* entre *Projectes* i *Entitats* amb el rol d'entitat coordinadora s'ha afegit a la taula de *Projectes* l'id de l'entitat que té com a coordinadora, actuant aquest nou atribut com a clau forana. El mateix passa amb les associacions entre *Projecte-Contacte*, *Projecte-User* i *Contacte-Entitat*. La taula amb la multiplicitat 1 guarda la clau primària de la taula amb la qual s'associa com a clau forana. Com a excepció, de l'associació *Projecte-User*

també s'ha afegit a la taula Projecte l'atribut `id_group`, ja que el grup d'usuaris al qual pertany un projecte es invariable mentre que un usuari podria canviar de grup.

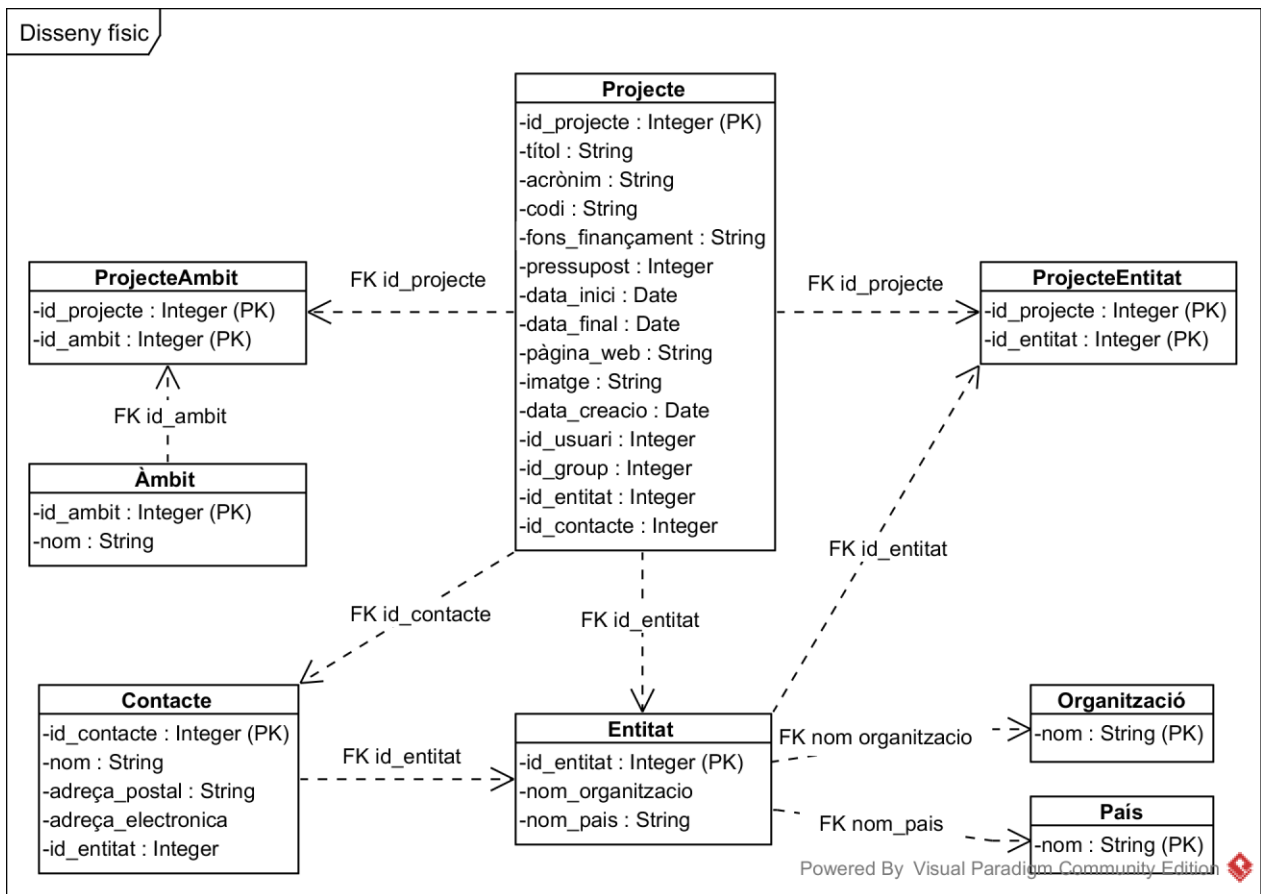


Figura 16. Disseny físic

Especificació

Organització (**nom**)

País (**nom**)

Entitat (**id_entitat**, `nom_organitzacio`, `nom_país`)

`Nom_organitzacio` fa referència a `nom` de la taula Organització

`Nom_pais` fa referència a `nom` de la taula País

Contacte (**id_contacte**, nom, adreça_electrònica, adreça_postal, id_entitat)

Id_entitat fa referència a id_entitat de la taula Entitat

Projecte (**id_projecte**, títol, acrònim, codi, fons_finançament, data_inici, data_final, pressupost, descripció, pàgina_web, logo, id_contacte, id_entitat, id_user, id_group, data_creació)

Id_contacte fa referència a id_contacte de la taula Contacte

Id_entitat fa referència a id_entitat de la taula Entitat

Àmbit (**id_ambit**, nom)

ProjecteEntitat (**id_projecte**, **id_entitat**)

Id_projecte fa referència a id_projecte de la taula Projecte

Id_entitat fa referència a id_entitat de la taula Entitat

ProjecteÀmbit (id_projecte, id_ambit)

Id_projecte fa referència a id_projecte de la taula Projecte

Id_ambit fa referència a id_ambit de la taula Àmbit

6. Implementació

6.1. Tecnologies i llenguatges emprats

Liferay

La tecnologia troncal al voltant la qual s'estructura el projecte és el CMS Liferay (Liferay 6.2). Aquest CMS té la capacitat que els desenvolupadors creïn mòduls (subsistemes dins d'un sistema més gran, que en el cas d'aquest projecte és RuralCat) personalitzats. A més a més, entre una gran quantitat de característiques, és compatible amb la major part de servidors d'aplicacions, bases de dades i sistemes operatius. D'altra banda, només suporta el llenguatge de programació Java, però en cap cas és un problema, ja que encaixa correctament amb el sistema a desenvolupar.

Java

Per programar tota la lògica del sistema s'ha utilitzat el llenguatge de programació Java [17], el qual era l'única opció possible, ja que Liferay no suporta més llenguatges. Java és un llenguatge interpretat i orientat a objectes. Les seves principals avantatges són el seu alt índex de re-utilització del codi, el fet de ser un llenguatge multiplataforma (sent una eina molt valorada per desenvolupar aplicacions distribuïdes) i la facilitat per programar que ofereix.

Hibernate

Juntament amb Java, per agilitzar i realitzar les consultes i les operacions a la base de dades s'ha utilitzat l'eina de mapeig d'objectes relacionals Hibernate [18], lliure i de codi obert. Aquesta eina permet simplificar les tasques relacionades amb la persistència de les dades i dotar les aplicacions de portabilitat entre SGBDs. A més a més, també disposa d'un llenguatge propi per realitzar consultes a la base de dades (Hibernate Query Language, HQL) i facilitats per a la recuperació de les dades.

Spring Web MVC

Spring Web MVC [19] és un framework de codi obert per Java que facilita la utilització del patró de programació Model-Vista-Controlador (MVC). Està dissenyat per tal que un controlador central proporciona un algorisme compartit, mentre que el treball real es realitza en components delegats configurables.

HTML/CSS

Per dissenyar la interfície web és necessari utilitzar un llenguatge de marques, no s'ha de confondre amb llenguatge de programació, ja que no són el mateix. Un llenguatge de programació (Java en el cas d'aquest projecte) té funcions aritmètiques i variables, mentre que un llenguatge de marques únicament conté marques que estructurin la interfície amb etiquetes que defineixen l'estil. En aquest sistema s'ha utilitzat el llenguatge HTML (HyperText Markup Language), el qual és el més utilitzat en entorns web.

El problema principal d'HTML, si s'utilitza sense cap mena de llenguatge de disseny gràfic, és que ofereix un aparença molt rudimentària i poc atractiva visualment. Per tant, es fa necessari utilitzar un llenguatge que proporcioni el disseny a la interfície, en el cas d'aquest projecte, CSS (Cascading Style Sheets). Disposem de tres maneres diferents per tal d'associar el disseny CSS amb els components HTML:

- Podem definir els estils en el mateix codi HTML (*Inline*). D'aquesta manera podem afegir estils molt ràpidament en curt termini, però ens obliga a definir els estils per a cada element (inclús si els estils són els mateixos). Per tant, aquesta manera de treballar afavoreix en gran mesura la duplicació de codi, fent molt feixuga la tasca de mantenir el codi a mig i a llarg termini (especialment en projectes grans).
- Una segona opció és definir els estils dintre el codi HTML però independent dels elements. D'aquesta manera, elements amb el mateix estil tindran la mateixa classe (agrupació d'estils CSS) i seran molt més mantenibles. Tot i això seguim creant duplicació de codi, ja

que cada interfície haurà de definir els seus estils, i aquests solen ser idèntics o molt similars dins un mateix sistema.

- L'última opció és definir els estils en un document a part. D'aquesta manera un mateix document pot ser accessible des de diverses interfícies HTML i ens permet eliminar qualsevol duplicació de codi (permetent una mantenibilitat molt eficient). A mig i a curt termini aquesta opció és la més viable tot i que sol ser lenta a curt termini.

En aquest projecte s'ha utilitzat principalment la tercera opció (codi CSS en documents a part), tot i que en algun cas particular (quan un estil s'utilitza només en un sol element) s'ha utilitzat l'opció *inline*. A més a més, s'ha utilitzat la convenció de nomenclatura de classes BEM (Block Element Modifier), obtenint així una major modularitat, reutilització i facilitat de comprensió del codi.

JavaScript

Per aconseguir una interfície dinàmica i que respongui davant la interacció entre el sistema i l'usuari s'ha utilitzat Javascript [20], un llenguatge interpretat i de tipatge dèbil (*weakly typed*). La principal avantatge d'utilitzar Javascript és la senzillesa d'ús i la seva rapidesa.

Maven

Gestionar les dependències i el cicle de vida de la construcció d'un projecte pot ser una tasca feixuga que alenteixi la velocitat de desenvolupament si no es gestiona correctament. Per evitar-ho s'ha utilitzat Maven (Apache Maven 3.5.4), el qual utilitza un Project Object Model (POM) per descriure el projecte de software a construir, les seves dependències d'altres mòduls i components externs i l'ordre de construcció dels elements. Té comandes per assolir objectius clarament definits com, per exemple, compilar el codi, desplegar el sistema al servidor o realitzar les proves.

SQL

Per administrar la base de dades s'ha utilitzat el llenguatge SQL (Structured Query Language), el qual és el llenguatge més utilitzat quan es disposa d'un model de dades relacional. Com ja s'ha mencionat, per realitzar les consultes des del sistema s'utilitza Hibernate (fent ús del seu llenguatge propi, Hibernate Query Language, molt semblant al SQL), però és necessari disposar

d'un llenguatge per crear i gestionar les diferents taules de la base de dades del sistema. Per tant, tot i tenir una eina de software per agilitzar el control de la base de dades, es segueix fent necessari disposar de SQL.

TortoiseSVN

TortoiseSVN (TortoiseSVN 1.11.0) és un client de Subversion, implementat com una extensió de la consola de Windows. S'ha utilitzat per tenir un control de les versions del sistema. Com que el projecte ha estat desenvolupat únicament per una sola persona, s'ha decidit no utilitzar cap metodologia de branching.

6.2. Eines de desenvolupament

Eclipse

Per la programació íntegra del sistema s'ha utilitzat l'IDE (Integrated Development Environment) Eclipse (Eclipse Java Photon 4.8.0), el més utilitzat en programar en Java. La principal avantatge d'aquest IDE és que disposa de nombrosos *plugins* que afegeixen funcionalitats, entre els quals s'han utilitzat tres diferents: un primer *plugin* de Tomcat (Eclipse Tomcat Plugin 9.3.1) per poder gestionar el servidor local per desenvolupar el sistema, un segon de Maven (Maven Integration for Eclipse 1.5) per poder realitzar totes les accions de Maven des de l'Eclipse i, en últim lloc, un *plugin* de Liferay (Liferay IDE 3.1.2) el qual adapta Eclipse a les necessitats de programar un mòdul per Liferay, transformant lleugerament l'IDE.

MySQL

Per administrar la connexió amb la base de dades i poder realitzar les consultes i operacions SQL s'ha utilitzat el SGBD (Sistema de Gestió de Bases de Dades) MySQL (MySQL Workbench 6.3.10). El principal avantatge d'aquesta eina és la rapidesa amb què efectua les consultes i operacions SQL.

7. Proves

Durant el desenvolupament del projecte s'han realitzat paral·lelament un seguit de proves per assegurar-se que el funcionament era l'esperat i no canviava a l'introduir noves funcionalitats. Les proves efectuades es poden dividir en dues categories: proves unitàries i proves d'execució.

Les proves unitàries tenen la funció de comprovar el correcte funcionament de les funcionalitats de forma aïllada, únicament el software. Són unes proves que, de forma individual, comproven una part molt concreta i reduïda del sistema (amb un rang d'escenaris que puguin cobrir els casos més comuns i els més extrems), però que de forma global cobreixen tota la lògica del sistema. Aquestes proves es poden executar de forma molt ràpida, sent possible comprovar que es compleixen cada cop que es modifica o s'afegeix una funcionalitat. Tot i això, aquestes proves no garanteixen el correcte funcionament del sistema en la seva totalitat, ja que no tenen en compte els elements externs com, per exemple, la interacció amb la base de dades i amb els usuaris. Per realitzar aquestes proves s'ha utilitzat JUnit, una biblioteca de Java per a la realització de les proves unitàries.

Per garantir el correcte funcionament del sistema en la seva totalitat s'han realitzat unes proves d'execució, les quals consisteixen a comprovar que el comportament és correcte quan s'interactua amb l'usuari, amb el servidor i amb la base de dades. Per tant, aquestes proves es realitzen de manera manual pel desenvolupador, i s'ha de procurar que cobreix tots els escenaris possibles. Com que són unes proves que requereixen cert temps només es realitzen en finalitzar cada iteració de la fase de construcció, al final de la qual s'obté un producte software complet, que pot ser provat de forma funcional. Si les proves es realitzen correctament, es dona per finalitzada la iteració, en cas contrari es busca la font de l'error i es tornen a realitzar les proves.

8. Aspectes Legals

8.1. Lleis aplicables al projecte

Aquest projecte essencialment serveix per fer difusió de projectes finançats per la Unió Europea i relacionats amb el món, per tant, qualsevol possible llei aplicable al projecte haurà de ser referent a aquest fet. La informació pròpia dels projectes no està subjecte a cap llei, ja que és aquests són públics. Únicament s'ha de tenir en compte no modificar la informació en introduir-la al sistema, però, com que la introduiran els mateixos responsables, RuralCat no s'ha de preocupar per aquest problema. El mateix passa amb la informació relativa a les entitats (organitzacions o empreses). El que si pot tenir rellevància legal, és la informació sobre el contacte dels projectes, la qual no deixa de ser informació personal i per tant es regeix per la Llei orgànica de protecció de dades de caràcter personal [21]:

“Es regeix per aquesta Llei orgànica qualsevol tractament de dades de caràcter personal quan el tractament s'efectuï en territori espanyol, en el marc de les activitats d'un establiment del responsable del tractament.” Article 2.a

I tal com defineix la mateixa llei la informació sobre les persones de contacte pot ser qualificada d'informació personal:

“Als efectes d'aquesta Llei orgànica, s'entén per Dades de caràcter personal: qualsevol informació referent a persones físiques identificades o identificables.” Article 3.a

Aquest projecte únicament utilitzaran les dades personals del contacte per a fer difusió del projecte tal com s'indica en la Llei orgànica:

“Les dades de caràcter personal només es poden recollir per ser tractades, així com sotmetre-les a aquest tractament, quan siguin adequades, pertinents i no excessives en relació amb l'àmbit i les finalitats determinades, explícites i legítimes per a les quals s'han obtingut.” Article 4.a

Tot i això el fet d'utilitzar aquestes dades no requereix avisar a l'afectat, ja que en primer lloc en ser la informació sobre els projectes pública significa que la persona de contacte ja ha donat prèviament el seu consentiment, i en cas de no ser així, el responsable seria la persona o l'entitat que publica la informació al sistema. En segon lloc, les dades personals a mostrar són molt escasses i profundament lligades a la resta de la informació. Per tant la publicació d'aquestes dades juntament amb la resta d'informació és prou justificada per a no avisar a l'afectat tal com estableix la Llei orgànica:

“El consentiment a què es refereix aquest article pot ser revocat quan hi hagi una causa justificada per fer-ho i no se li atribueixin efectes retroactius.” Article 6.3

En conclusió, només les dades sobre el contacte del projecte estan afectades per alguna llei, tot i això, a causa del volum de les dades (que és molt petit i de poca importància) i a la manera que el sistema les utilitza (únicament per la seva difusió com a informació associada a un grup d'informació major), no es pot generar cap problema legal.

8.2. Llicències

En tot moment la política del projecte ha estat utilitzar productes amb llicències lliures per tal d'abaratir costos. Tot i això és important desglossar, una a una, totes les llicències dels productes utilitzats per poder tenir un control sobre les mateixes:

Nom del producte	Llicència	Tipus de llicència
Eclipse	EPL	Lliure
Hibernate	GPL	Lliure, Copyleft
Liferay	LGPL	Lliure, Copyleft

MySQL	GPL	Lliure, Copyleft
Spring	Apache Licence	Lliure
TortoiseSVN	GPL	Lliure, Copyleft

Taula 8. Llicències

Totes les llicències mencionades a la taula X són de programari lliure. Per tant, permeten a l'usuari del programari plena llibertat d'ús per a qualsevol propòsit, distribuir-lo, modificar-lo, i distribuir versions modificades del programari. En les llicències copyleft, també es protegeix al producte de ser distribuït amb una llicència diferent, és a dir, garanteix que el software es mantingui sempre lliure, però no afecta en cap cas a l'ús del software.

No s'han tingut en compte les llicències relacionades amb llenguatges de programació o similars com Java,HTML,CSS i Javascript, ja que les llicències sota les que s'emparen protegeixen la implementació dels llenguatges i no el seu ús. Utilitzar un d'aquests llenguatges és com parlar o escriure un idioma, no pot ser restringit.

9. Seguiment del projecte

9.1. Canvis de planificació i pressupost

Segons la planificació del projecte el temps de desenvolupament estava previst a 510 hores, però durant el transcurs del mateix s'ha vist que l'execució real ha variat respecte a la previsió inicial. Hi ha hagut dues variacions principals: una primera variació que ha reduït els temps previstos i una segona que ha obligat a ajornar temporalment el projecte, augmentant així el temps total però no el temps de feina real.

El primer canvi respecte a la planificació inicial redueix lleugerament el temps de feina per al desenvolupament del projecte. Aquesta desviació es deu al fet que no es van estimar de manera prou precisa les hores que podria durar el desenvolupament del projecte. Tot i que pot ser per diversos factors, segurament el principal és la manca d'experiència en la planificació d'un projecte real. En total aquesta desviació disminueix el nombre d'hores en un 10% aproximadament. A més a més, la fase de Transició s'ha realitzat a mitges, ja que el client s'ha demorat a donar el vist i plau per a desplegar el sistema al portal. Per tant, el desplegament no s'ha realitzat i tampoc s'ha pogut obtenir feedback dels usuaris.

Tasca	Hores previstes	Pressupost previst (€)	Hores reals	Pressupost real (€)
Incepció	60	480	50	400
Elaboració	120	960	100	800
Construcció	270	2160	250	2000
Transició	60	480	40	480

Total	510	4080	440	3520
--------------	-----	------	-----	------

Taula 9. Desviacions en la planificació (hores i pressupost)

La segona desviació té un caràcter diferent de la primera, no afegeix o disminueix hores a la càrrega de treball prevista però modifica les dates previstes. Aquesta desviació es deu a dos motius. En primer lloc no tot el temps treballat a l'empresa és temps dedicat al projecte (ja que també es dedica temps a les pràctiques, alienes al projecte). En realitzar la planificació inicial no es va tenir en compte això i per tant les dates previstes es veuen afectades. En segon lloc el projecte es va ajornar, ja que el conveni entre Adasa i la UPC per realitzar el Treball de Fi de Grau no va estar llest en la data indicada (16 de setembre) i fins que el conveni no va estar llest no es va poder reprendre el projecte. Aquesta aturada va durar entre 3 i 4 setmanes, gairebé un més. Degut a això les dates previstes novament es veuen afectades.

Tasca	Data inici prevista	Data final prevista	Data inici real	Data final real
Incepció	02/07/18	13/07/18	02/07/18	18/07/18
Elaboració	16/07/18	10/08/18	19/07/2018	24/08/2018
Construcció	13/08/18	12/10/18	27/08/2018	07/12/2018
Transició	15/10/18	26/10/18	10/12/2018	22/12/2018

Taula 10. Desviacions en la planificació (dates)

10. Sostenibilitat

Dia rere dia, realitzar correctament l'informe de sostenibilitat es fa més important, ja que estima quin impacte social, ambiental i econòmic tindrà el projecte en la seva totalitat. Aquest capítol presenta aquest informe, el qual s'ha realitzat donant resposta a una sèrie de preguntes ja definides [annex], basat en la matriu de la taula 11. L'anàlisi de sostenibilitat d'un projecte es divideix en tres parts, identificades per les columnes de la matriu:

- El projecte posat en producció (PPP), que inclou
- la planificació el desenvolupament i la implementació del projecte.
- La vida útil del projecte, que comença un cop implantat el projecte i acaba amb el seu desmantellament.
- Els riscos inherents al projecte durant la seva construcció, vida útil i desmantellament.

	PPP	Vida útil	Riscos
Ambiental	Consum de disseny	Petjada ecològica	Ambientals
Econòmic	Factura	Pla de viabilitat	Econòmics
Social	Impacte personal	Impacte social	Socials

Taula 11. Model matriu de sostenibilitat

Cada una de les columnes s'ha d'analitzar des de tres punts de vista (identificats per les files): l'ambiental, l'econòmic i el social, les tres dimensions de la sostenibilitat. El significat de cada una de les cel·les de la matriu és el següent:

- Ambiental/PPP: representa l'impacte sobre el medi ambient al llarg de la realització del TFG. La petjada ecològica es pot mesurar, per exemple, en kWh i en tones d'emissions provocats pel desenvolupament del TFG.
- Ambiental/Vida útil: representa la petjada ecològica que tindrà el projecte durant tota la seva vida útil, és a dir, des que està implementat fins que es desmantella. Es pot mesurar en els mateixos paràmetres que la cel·la anterior.
- Ambiental/Riscos: representa el conjunt d'eventualitats que podrien causar que l'impacte ambiental del projecte sigui més negatiu del previst.
- Econòmic/PPP: representa el consum de recursos (materials i humans) realitzat durant el desenvolupament de tot el projecte i el cost addicional d'aquests recursos. Seria l'equivalent a la factura que es cobraria a un potencial client del projecte i requereix una planificació detallada del TFG.
- Econòmic/Vida útil: representa el pla de viabilitat de projecte, plantejat de forma simplificada.
- Econòmic/Riscos: representa el conjunt d'eventualitats que podrien causar que el projecte tardés més del previst, o inclús mai, en aconseguir la viabilitat.
- Social/PPP: representa l'impacte que la realització del projecte ha tingut sobre les persones que han treballat en el mateix.
- Social/Vida útil: representa l'impacte que la posada en marxa del projecte tindrà sobre els diferents col·lectius relacionats, ja sigui d'una forma directa o indirecta.
- Social/Riscos: representa el conjunt d'eventualitats que podrien causar que l'impacte social del projecte sigui més negatiu del previst.

10.1. Dimensió ambiental

Per realitzar l'informe de la dimensió ambiental, s'ha quantificat l'impacte ambiental del projecte en kWh. A l'igual que en els costos indirectes, per realitzar l'anàlisi d'aquest impacte únicament s'han tingut en consideració les despeses energètiques generades únicament per aquest projecte. El motiu és que el consum de calefacció o aire condicionat (depenent l'estació de l'any) i de la il·luminació de les instal·lacions del lloc de treball ja es generava abans de l'inici del projecte i en

cap cas es veu alterat. Per tant, l'únic consum generat per aquest projecte és el de l'equip informàtic utilitzat pel desenvolupador, el qual es pot desglossar en el consum del monitor i de l'ordinador.

En la següent taula poden veure el consum de cada dispositiu i quin consum diari i total té. El consum diari es calcula tenint en compte que un dia laborable del desenvolupador és de 6h. El total es calcula tenint en compte que, com s'ha descrit en la planificació inicial, el projecte té una durada de 85 dies laborables.

Motiu	Consum	Dia laboral (6h)	Total (85 dies)
Monitor (HP L1750)	30W	0,18kWh	15,3kWh
Ordinador (Dell Precision T1600)	320W	1,92kWh	163,2kWh
Total	350W	2,10kWh	178,5kWh

Taula 12. Consum energètic

Per reduir l'impacte generat s'han pres certes mesures. En primer lloc l'ordinador s'apaga totalment un cop s'abandona el lloc de treball, reduint així el consum a 0 durant la resta del dia. Si no es fes així, es podria deixar l'ordinador suspès com a alternativa (deixar-lo totalment encès no és una opció, ja que gairebé tots els ordinadors estan configurats per defecte a quedar en estat de suspensió després d'un període d'inactivitat), reduint el consum a 5W aproximadament [22]. Per tant, gràcies al fet d'apagar per complet l'ordinador s'aconsegueix un estalvi de 11,49kWh. Tot i això, és molt comú oblidar-se encesa la regleta de la pantalla, la qual té un consum de 4W. La segona mesura presa és apagar aquesta regleta, estalviant així el seu consum, 9,19kWh. En la següent taula podem veure els consums descrits i el total estalviat:

Estat	Consum	Dia (18h)	Cap de setmana (2 dies)	Total (85 dies + 16 caps de set.)
Ordinador Suspès	5W	0,09kWh	0,24kWh	11,49kWh
Regleta encesa	4W	0,072kWh	0,192kWh	9,192kWh
Total	9W	0,162kWh	0,432kWh	20,682kWh

Taula 13. Estalvi energètic

En cas de voler augmentar l'estalvi en el consum es podria plantejar l'opció d'adquirir material més eficient energèticament. El monitor de l'ordinador té un consum relativament baix, però l'ordinador no, ja que podria tenir un consum de 200W fàcilment. Adquirir un ordinador amb aquestes característiques energètiques suposaria un estalvi energètic enorme, de gairebé un 40% respecte a l'actual. Com que s'han utilitzat els mínims recursos possibles el consum no es podria reduir més.

Durant la vida útil del projecte els recursos que s'utilitzaran són el servidor on es desplega el sistema i la base de dades on s'emmagatzema tota la informació. Tot i això, al ser recursos compartits amb un projecte major, s'ha decidit no tenir-los en compte.

El que si es tindrà en compte, és el benefici ambiental indirecte que pot generar el projecte. Tal com s'ha explicat en la introducció, l'objectiu principal d'aquest projecte és la difusió de projectes finançats amb fons europeus i relacionats amb el món rural. La major part d'aquests projectes estan profundament lligats amb el medi ambient i sempre en un sentit positiu. Fer més visibles aquests projectes significa donar-li's suport i afavorir que tinguin més suport, tant popular com empresarial. Per tant, aquest projecte té un impacte positiu, tot i que molt reduït i de forma totalment indirecte, sobre la petjada ecològica global.

L'únic risc possible que fes augmentar la petjada ecològica és que la unió europea financés projectes amb una mala sostenibilitat ambiental. Però, degut a l'impacte mediàtic i social que té actualment l'ecologisme, és un escenari molt improbable.

10.2. Dimensió econòmica

Per realitzar aquest projecte s'ha realitzat una previsió dels costos abans de començar el projecte, la qual està en el capítol de gestió econòmica. Aquesta previsió ha resultat ser més costosa del que realment ha costat el projecte, ja que les hores de treball no han sigut tantes com s'havia previst a la planificació inicial. El motiu principal d'aquesta desviació respecte a la previsió inicial ha estat la inexperiència de realitzar una planificació temporal real. Tot i això la diferència total ha estat relativament acceptable, aproximadament un 10% del cost previst.

El cost total del projecte és, en la seva major part, el cost dels recursos humans, que en aquest cas és un únic desenvolupador. Per tal de reduir aquest cost es podria reduir el sou del treballador, opció no permesa pel conveni i que reduiria les condicions laborals del treballador (afectant negativament a la dimensió social), o reduir el número d'hores. L'única forma de poder reduir les hores seria contractar a un treballador amb més experiència, que pogués realitzar la feina amb més eficiència. El problema, però, és que un treballador més experimentat segurament requerirà un sou major. La resta del cost del projecte, els costos indirectes, són els generats pel consum elèctric, i que s'ha detallat en la dimensió anterior.

Els costos del projecte durant la seva vida útil podrien ser els generats per la despesa de consum del servidor i la base de dades. Tot i això, ja s'ha explicat que no es considerarà aquest consum propi del projecte, i per tant tampoc els costos de la despesa. El que sí pot tenir un cost durant la vida útil del projecte, és el manteniment que pot necessitar el sistema. Com que el projecte ha estat dissenyat per ser fàcilment mantenible i molt flexible amb l'addició de noves funcionalitats o amb la modificació de funcionalitat ja existents, el cost de manteniment no hauria de ser elevat. Per tant, al realitzar el projecte s'ha pensat amb visió a llarga durada i en reduir al màxim els possibles costos que es poden generar durant la vida útil del projecte.

La viabilitat econòmica d'aquest projecte només es pot veure afectada pels canvis en els costos del projecte. Els beneficis vénen donats pel contracte públic entre el Departament d'agricultura, ramaderia i pesca i Adasa, el qual estableix un pagament invariable (l'acordat en realitzar el contracte de RuralCat). Els costos poden augmentar en cas que les hores de treball augmentin, però en el seguiment del projecte ja s'ha vist que ha passat el contrari. Per tant la viabilitat del projecte en cap cas està en risc.

10.3. Dimensió social

En l'àmbit personal aquest projecte m'ha provocat una impressió positiva. El fet de gestionar un projecte des de l'inici fins a desplegar-lo proporciona una experiència molt enriquidora professionalment i una visió molt més global del que representa un projecte de software. En definitiva, l'oportunitat de treballar en un entorn real.

El principal beneficiari d'aquest projecte és el mateix client, el Departament d'agricultura i pesca, i la Unió Europea. En ser aquestes dues organitzacions de caràcter públic també podem contar entre els beneficiaris, tot i que en un grau molt menor, la població que comprenen aquestes organitzacions, especialment la situada a Catalunya i que està laboralment relacionada amb el món rural.

Com que l'impacte social d'aquest projecte és intrínsec a l'impacte que generen els projectes dels quals fa difusió, és possible que algun col·lectiu es vegi afectat de manera negativa, ja que qualsevol iniciativa, per poder beneficiar un col·lectiu o per poder millorar la dimensió ambiental, és molt probable que hagi que perjudicar un altre col·lectiu. Tot i això, s'espera d'organitzacions governamentals que els projectes que financin tinguin un major impacte positiu global que negatiu, i per tant, el mateix s'espera d'aquest projecte.

10.4. Matriu de sostenibilitat

	Dimensió econòmica	Dimensió ambiental	Dimensió social
PPP	8	6	8
Vida útil	7	8	7
Riscos	-4	-1	-1

Taula 14. Matriu de sostenibilitat

11. Conclusions i treball futur

11.1. Competències tècniques treballades i relació amb l'enginyeria del software

Aquest treball està englobat dins l'enginyeria del software, una de les cinc especialitats de l'enginyeria informàtica, centrada en el desenvolupament, manteniment i avaluació de serveis i sistemes software. En aquest capítol s'analitzarà si aquest projecte està relacionat amb aquesta especialitat i es justificaran els motius.

Per realitzar aquest projecte s'ha realitzat un anàlisi inicial important, constituït per un anàlisi del problema a resoldre, el plantejament d'una possible solució i la gestió temporal i econòmica que suposaria implementar aquesta solució. Per tant, aquest projecte es sustenta darrere un anàlisi que justifica la seva realització. A més a més, un cop vist el context del projecte, s'han extret i especificat els requisits necessaris per implementar el projecte de forma satisfactòria. Aquests coneixements estan fortament lligats a assignatures de l'especialitat com Gestió de Projectes de Software (GPS), i Enginyeria de Requisits (ER).

Igual que en l'assignatura Projecte d'Enginyeria del Software (PES), per englobar tot el flux de treball s'ha utilitzat una metodologia de treball, la qual ha estat ben definida des de l'inici del projecte. Gràcies a aquest marc de treball s'ha pogut estructurar la càrrega de treball del projecte de forma estructurada i amb flexibilitat per suportar possibles imprevistos.

Per a l'implementació del sistema s'han utilitzat totes les tècniques i bones pràctiques possibles per tal de garantir la millor qualitat possible del software resultant. Aquestes tècniques, apreses en gran mesura durant l'assignatura d'Arquitectura del Software (AS), inclouen els patrons utilitzats, la correcta utilització de l'arquitectura escollida i una correcta especificació del comportament del sistema i del model de les dades.

Finalment, per tal de dissenyar la base de dades utilitzada en aquest projecte, s'han aplicat coneixements de l'assignatura Disseny de Base de Dades (DBD).

Com es pot apreciar, al ser un projecte que engloba tot el cicle de vida del desenvolupament, resulta estar correctament relacionat amb l'enginyeria del software i, en diferents mesures, amb totes les branques de coneixement o competències que componen l'especialitat.

La relació amb les diferents competències de l'especialitat es va definir prèviament a iniciar el projecte, a continuació s'enumeren, juntament amb el grau de profunditat amb el que es treballen i una justificació del motiu de la seva tria:

- **CES1.1:** Desenvolupar, mantenir i avaluar sistemes i serveis de software complexos i/o crítics. [En profunditat]: aquesta competència ha estat escollida perquè el sistema resultat d'aquest projecte, que s'ha desenvolupat d'inici a fi, respon a una necessitat real. S'ha escollit el nivell *en profunditat*, ja que s'han hagut de realitzar totes les tasques del desenvolupament del projecte.
- **CES1.2:** Donar solució a problemes d'integració en funció de les estratègies, dels estàndards i de les tecnologies disponibles. [Una mica]: com que el sistema desenvolupat s'ha hagut d'integrar dins d'un sistema major ja existent, RuralCat, s'ha escollit aquesta competència. El nivell de profunditat triat, *una mica*, és degut que aquesta integració no ha sigut complexa de realitzar (degut a que s'han utilitzat les mateixes tecnologies en ambdós sistemes).
- **CES1.5:** Especificar, dissenyar, implementar i avaluar bases de dades. [En profunditat]: a l'iniciar el projecte, es va descobrir que la base de dades ja existent per guardar la informació no complia cap requisit de qualitat ni cap estàndard. Per tant, va ser necessari reestructurar completament la base de dades, amb l'afegit de mantenir la coherència de les dades ja existents. S'ha triat el nivell *en profunditat*, ja que aquesta competència ha sigut molt important durant la realització del projecte i ha sigut molt treballada.
- **CES1.7:** Controlar la qualitat i dissenyar proves en la producció de software. [Una mica]: durant el desenvolupament d'aquest projecte s'han realitzat un seguit de proves per asse-

gurar el correcte funcionament del sistema, tal com s'ha descrit en el capítol 7 d'aquest document. A causa d'això s'ha inclòs aquesta competència amb el nivell de profunditat *una mica*.

- **CES2.1:** Definir i gestionar els requisits d'un sistema software. [Bastant]: a l'iniciar el projecte, tot i que l'idea general ja estava plantejada, no hi havia cap requisit especificat que el software resultant hagués de complir. Per tant, s'han hagut de definir i analitzat tots els requisits per poder, posteriorment, implementar correctament el sistema. S'ha escollit el nivell de profunditat *bastant*, ja que, tot i que s'ha especificat tots els requisits, no s'ha establert mai un contacte directe amb el client.

11.2. Conclusions personals

Aquest projecte ha estat una experiència personal molt positiva. Durant la realització de la carrera s'ha simulat, en diferents assignatures, el desenvolupament d'un sistema de forma total o parcial (centrant-se en una fase concreta, com per exemple, l'extracció dels requisits en el cas de l'assignatura Enginyeria dels requisits), tot i això, tot i ser experiències enriquidores que proporcionen una visió pràctica, no deixen de ser un entorn simulat i que no reflecteix els problemes que poden sorgir en un projecte real. Per aquest motiu, treballar amb un entorn real, on poden sorgir una gran diversitat de inconvenients, et mostra que el més probable és que es produeixin algun imprevist que haurà de ser gestionat.

Dels projectes realitzats durant el grau, els que comprenien una fase concreta del procés de desenvolupament no et permetien veure les conseqüències (positives o negatives) de les decisions preses en futures fases. En el cas dels projectes realitzats en la seva totalitat, com el realitzat a l'assignatura Projecte de l'Enginyeria del Software, a l'estar involucrats diversos desenvolupadors, no tenies una visió global de tot el projecte, ja que realitzaves uns rols concrets i unes tasques específiques. En el cas d'aquest projecte, el qual s'ha desenvolupat íntegrament per una sola persona, s'ha pogut veure els efectes de les decisions preses en les primeres fases del desenvolupament del projecte, així com una visió total del mateix.

En conclusió, crec que aquesta experiència ha permès englobar i aplicar tots els coneixements apresos (en diverses mesures) durant el grau i durant l'especialitat. A més a més, m'ha proporcionat l'oportunitat d'obtenir experiència laboral, que podrà ser molt beneficiosa en el futur.

11.3. Treball futur

Tot i que el sistema s'ha desenvolupat amb èxit i ha complert tots els objectius plantejats, no s'ha arribat encara a desplegar al servidor de producció (el que és accessible pels usuaris). Per tant, en un futur molt proper, quan el client ho indiqui, s'haurà de realitzar aquest desplegament i crear les condicions adequades perquè el projecte pugui funcionar. En primer lloc s'hauran d'assignar els rols als usuaris que pertoquin per tal de permetre'ls l'accés al sistema. En segon lloc s'hauran de crear els grups d'usuaris i afegir-li'ls els usuaris. Finalment caldrà afegir els noms de les entitats i els països. Totes aquestes tasques no s'han pogut realitzar encara perquè és necessari que la informació requerida per dur-les a terme la proporcioni el client.

Un cop el sistema hagi estat desplegat i estigui en funcionament caldrà rebre el feedback dels usuaris. Si es trobes algun error caldria arreglar-lo, tot i que, gràcies a les proves realitzades haurien de ser inexistents o molt reduïts. En cas que plantegin alguna millora o desitgin modificar alguna funcionalitat s'haurà d'analitzar els nous requisits i estudiar la viabilitat de portar-los a terme.

12. Referències

- [1] Adasa, «Water and Environment Technology,» [En línia]. Available: <http://www.adasasistemas.com/esp/home.php>. [Últim accés: 20 Setembre 2018].
- [2] Comsa Corporation. [En línia]. Available: <http://www.comsa.com/web/comsawp/home>. [Últim accés: 20 Setembre 2018].
- [3] RuralCat, «Benvinguda,» [En línia]. Available: <https://ruralcat.gencat.cat/web/guest/benvinguda>. [Últim accés: 20 Setembre 2018].
- [4] Liferay, «Digital Experience Software Tailored to Your Needs,» [En línia]. Available: <https://www.liferay.com/es/home>. [Últim accés: 20 Setembre 2018].
- [5] Idescat, «Població ocupada. Per sectors d'activitat,» [En línia]. Available: <https://www.idescat.cat/indicadors/?id=conj&n=10204>. [Últim accés: 6 Novembre 2018].
- [6] WordPress, «Create a Free Website or Blog,» [En línia]. Available: <https://wordpress.com/>. [Últim accés: 20 Setembre 2018].
- [7] Joomla Content Managment System, «Try it for free!,» [En línia]. Available: <https://www.joomla.org>. [Últim accés: 22 Setembre 2018].
- [8] Drupal, «Open Source CMS,» [En línia]. Available: <https://www.drupal.org>. [Últim accés: 22 Setembre 2018].
- [9] Viquipèdia, «Model de desenvolupament en cascada,» [En línia]. Available: https://ca.wikipedia.org/wiki/Model_de_desenvolupament_en_cascada. [Últim accés: 24 Setembre 2018].

- [10] Wikipedia, «Unified Process,» [En línia]. Available: https://en.wikipedia.org/wiki/Unified_Process. [Últim accés: 24 Setembre 2018].
- [11] Wikipedia, «Program evaluation and review technique,» [En línia]. Available: https://en.wikipedia.org/wiki/Program_evaluation_and_review_technique. [Últim accés: 28 Setembre 2018].
- [12] FIB, «Pràctiques en empresa,» [En línia]. Available: <https://www.fib.upc.edu/ca/empresa/practiques-en-empresa>. [Últim accés: 14 Octubre 2018].
- [13] Indeed, «Sueldos en Jefe de proyecto en España,» [En línia]. Available: <https://www.indeed.es/salaries/Jefe-de-proyecto-Salaries>. [Últim accés: 14 Octubre 2018].
- [14] Indeed, «Sueldos en Analista funcional senior en España,» [En línia]. Available: <https://www.indeed.es/salaries/Analista-funcional-senior-Salaries>. [Últim accés: 14 Desembre 2018].
- [15] Indeed, «Sueldos en Arquitecto/a java en España,» [En línia]. Available: <https://www.indeed.es/salaries/Arquitecto/a-java-Salaries>. [Últim accés: 14 Desembre 2018].
- [16] Indeed, «Sueldos en Programador/a senior en España,» [En línia]. Available: <https://www.indeed.es/salaries/Programador/a-senior-Salaries>. [Últim accés: 14 Octubre 2018].
- [17] Oracle, «The Java language specifications,» [En línia]. Available: <https://docs.oracle.com/javase/specs/jls/se8/jls8.pdf>. [Últim accés: 4 Desembre 2018].
- [18] RedHat, «Hibernate,» [En línia]. Available: <http://hibernate.org>. [Últim accés: 4 Desembre 2018].

- [19] Spring Framework, «Web on Servlet Stack,» [En línia]. Available: <https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html>. [Últim accés: 4 Desembre 2018].
- [20] Wikipedia, «JavaScript,» [En línia]. Available: <https://ca.wikipedia.org/wiki/JavaScript>. [Últim accés: 5 Desembre 2018].
- [21] Departament d. Justícia, «Llei orgànica de protecció de dades de caràcter personal,» [En línia]. Available: http://administraciojusticia.gencat.cat/web/.content/documents/arxius/lo15_1999lopdcg.pdf. [Últim accés: 15 Desembre 2018].
- [22] OCU, «Vivienda y energia,» [En línia]. Available: <https://www.ocu.org/vivienda-y-energia/nc/calculadora/consumo-en-stand-by>. [Últim accés: 28 desembre 2018].
- [23] Wikipedia, «Front and Back ends,» [En línia]. Available: https://en.wikipedia.org/wiki/Front_and_back_ends. [Últim accés: 29 Setembre 2018].
- [24] HardZone, «Calcular consumo energia PC,» [En línia]. Available: <https://hardzone.es/2015/03/31/cuanto-cuesta-la-electricidad-que-consume-tu-pc/>. [Últim accés: 6 Octubre 2018].